中华学习机

CHINA EDUCATIONAL COMPUTER

- 程序剖析与编程技巧
- 苹果园地
- 实用程序
- 辅助教学与研究
- 专家论坛

4
1990



WEIFANG COMPUTER COMPANY

潍坊計算機公司

無線電話機



FTC—5203型、FTH—7005型400 MHz頻段頻率合成袖珍式無線電話機。台式充電器。

FM2516U型400 兆赫頻率合成全雙工無線電話機。PE—100型編碼器。WYZ—10A穩壓電源。

中华学习机 (原〈苹果园〉)

1990. No. 4 总 No. 4

目 录

编辑出版： 《中华学习机》编辑部

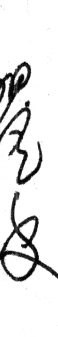
地址：山东潍坊东风大街 60 号
电话：226921—321、邮码：261041
FAX：(0536)225397
Telex：324304 CCWF CN

印刷： 潍坊计算机公司激光照排实验印刷厂

山东省报刊特许证第 012 号
工本费：0.90 元

刊名题字：李铁映

自办发行 常年订阅



BASIC语言中动态数组的实现

浙江温州师范学院　陈宣钱

摘要:数组在程序设计中应用得极为广泛,由于BASIC语言未能动态地申请分配或释放数组元素的存贮单元,程序设计者往往是根据大慨情况给数组开辟足够大的存贮空间,不能合理的使用内存。本文提出一种建立一维动态数组的一个方案。

一、引言

使用BASIC语言进行程序设计时,对引用的数组必须用DIM语句加以说明,以确定数组所占的存贮空间。然而,程序执行过程中往往不可预知地申请分配或释放数组的部分存贮空间,即增加数组所占的存贮单元或减少数组所占的存贮单元。DIM语句不能对数组进行重复性说明,也就不可能随着用户程序临时申请,给数组分配更多的存贮单元;数组中无用的存贮单元也无法释放。下面就从数组的内部结构出发,探讨一维数组的动态实现。

二、数组的结构和存贮

数组是在程序运行过程中建立和使用的,每一个数组都由一个称为表题的数据结构来标识。如图1所示。表题是根据程序中数组说明的先后依次存放在数组区内,数组的内容紧跟在相应数组的表题之后。实型占5个字节,整型占2个字节,字符串型占3个字节。一维数组只有一个下标表题就占7个字节。其中数据项数组长度反映该数组在内存所占的字节数;数据项下标值则反映数组可使用的数组元素的个数。数组一经DIM语句说明,这两个数据项的内容也就确定了。从以上分析看出,要建立动态数组,就必须更改表题的内容。

数组表题

表题	说明
	数组名称
低八位	数组长度
高八位	
	维　数
高八位	第n维下标值
低八位	
⋮	
高八位	第1维下标值
低八位	

图一

三、动态数组的实现

在表题的数据结构中,反映数组大小的有两个数据项:数组长度和下标值。为了实现申请分配或释放数组的存贮空间,就要求在程序运行时,每当引用一个数组元素,先把该数组的数组长度项增加一个数据类型的长度,下标值加1,为该数组元素分配相应的存贮单元;每当废除一个数组元素,数组长度项就减去一个数据类型的长度,下标值减1,释放该数组元素所占的存贮单元。如此进行,就可以做到不断地申请分配或释放数组的存贮空间。

有了以上的说明,就可以建立动态数组,具体作法是:先给数组说明一个数组元素,初始化程序中所有变量的值,以确定数组表题在内存的地址,并由此计算数据项数组长度和下标值在内存的地址。然后根据程序执行的需要,当申请一个数组元素时,就先调子程序为其分配存贮单元;当释放一个数组元素时,就调子程序释放数组元素所占的存贮单元。这样就实现了动态地申请或释放数组的存贮单元。

程序中用来实现动态的数组,最好是用户最后说明的一个数组。若第i个数组实现动态,则随着第i个数组所占存贮单元的变化,直接影响其后若干个数组的内容,解决方法必须相应移动第i个数组之后其它数组的表题及其内容。第i个数组的表题地址则指向数组区中前i-1个数组的长度和。

四、动态数组的应用

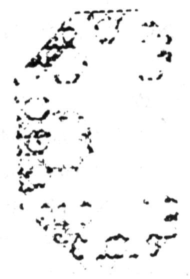
递归调用是高级语言的一种重要功能,尽管BASIC语言不支持递归调用,但人们借助数组还是实现了递归调用。由于递归调用的次数往往随着程序的运行而急剧上升,究竟给数组开辟多大的存贮空间,难以确定。然而,利用动态数组就能够很好的解决这个问题。

下面是一个用动态数组实现递归定义图形"C"曲线的BASIC程序和运行结果。

```
5  HOME : DIM ST(0)
10  I=1 : PI=3.14159
15  X0=120 : Y0=120:NX=0 : NY=0
20  INPUT"ENTER LENGTH AND ANGLE=";LN,
    AN
30  HGR2 : HCOLOR=3 : HPLOT X0,Y0
40  CALL 768
50  IF LN<=10 THEN 100
60  CALL 803 : CALL 803
70  ST(I)=LN : ST(I+1)=AN : I=I+2
```

```
80  LN=LN/4 : AN=AN+PI/4
90  GOTO 50
100  GOSUB 200
105  IF I<3 THEN END
110  I=I-2
130  LN=ST(I) : AN=ST(I+1)
140  LN=LN/4 : AN=AN-PI/4
150  CALL 834 : CALL 834
160  GOTO 50
200  NX=X0+LN * COS(AN)
210  NY=Y0-LN * SIN(AN)
220  HPLOT TO NX,NY
230  X0=NX : Y0=NY :  RETURN
```

运行结果：



图二

程序中调用的三个汇编语言子程序：

子程序 1

```
0300—  18          CLC
0301—  AD 6B  00   LDA  $006B
0304—  69 02       ADC  #$02
0306   8D 06  00   STA  #$006
0309—  AD 6C  00   LDA  $006C
030C—  69 00       ADC  #$00
030E—  8D 07  00   STA  $0007
0311—  18          CLC
0312—  AD 6B  00   LDA  $006B
0315—  69 05       ADC  #$05
0317—  8D 08  00   STA  $0008
031A—  AD 6C  00   LDA  $006C
031D—  69 00       ADC  #$00
031F—  8D 09  00   STA  $0009
0322—  60          RTS
```

子程序 2：

```
0323—  18     CLC
0324—  A2 00  LDX  #$00
0326—  A0 01  LDY  #$01
0328—  A1 06  LDA  ($06,X)
032A—  69 05  ADC  #$05
032C—  81 06  STA  ($06,X)
032E—  B1 06  LDA  ($06),Y
0330—  69 00  ADC  #$00
0332—  91 06  STA  ($06),Y
0334—  18     CLC
0335—  B1 08  LDA  ($08),Y
0337-  69 01  ADC  #$01
0339—  91 08  STA  ($08),Y
033B—  A1 08  LDA  ($08,X)
033D—  69 00  ADC  #$00
033F—  81 08  STA  ($08,X)
0341—  60     RTS
```

子程序 3

```
0342—  38     SEC
0343—  A2 00  LDX  #$00
0345—  A0 01  LDY  #$01
0347—  A1 06  LDA  ($06,X)
0349—  E9 05  SBC  #$05
034B—  81 06  STA  ($06,X)
034D—  B1 06  LDA  ($06),Y
034F-  E9 00  SBC  #$00
0351—  91 06  STA  ($06),Y
0353—  38     SEC
0354—  B1 08  LDA  ($08),Y
0356—  E9 01  SBC  #$01
0358—  91 08  STA  (#08),Y
035A—  A1 08  LDA  ($08,X)
035C-  E9 00  SBC  #$00
035E—  81 08  STA  ($08,X)
0360—  60     RTS
```

子程序 1 确定表题中数据项数组长度和下标值项在内存的地址。

子程序 2 分配一个数组元素所需的存贮单元。

子程序 3 释放一个数组元素所占的存贮单元。

程序执行后，除 **DIM** 中说明的一个数组元素外，其它给数组元素分配的存贮单元全部释放。

五、结束语

动态数组的实现，对用户来说实现了按需分配存贮单元，避免了存贮单元的浪费，达到节约内存的作用，但所有这些都是以时间为代价的。

一种简易汉化DOS的方法

哈尔滨船舶工程学院　朴一兵

CEC—I型中华学习机的用户大多数使用DOS3.3磁盘操作系统，虽然该系统在CEC—I机的中文状态下，比较满意地工作，但因其输出信息全是英文，给不懂英文的用户带来了很大的不便。本文给出能使DOS3.3的出错信息汉化的一个简单方法。在监控状态下，只需将下面给出的数据准确无误地输入，就可使出错信息全部用汉字给出。

首先简述一下原理。分析DOS3.3的结构可知，DOS3.3的出错信息是存放在$A974～$AA3D的内存单元里，当产生错误后，到相应的地址中取出信息显示在屏幕上。出错信息与相应的内存单元的关系是：

LANGUAGE NOT AVAILABLE	$A974～$A989
RANGE ERROR	$A98A～$A994
WRITE PROTECTED	$A995～$A9A3
END OF DATA	$A9A4～$A9AE
FILE NOT FOUND	$A9AF～$A9BC
VOLUME MISMATCH	$A9BD～$A9CB
I/O ERROR	$A9CC～$A9D4
DISK FULL	$A9D5～$A9DD
FILE LOCKED	$A9DE～$A9E8
SYNTAX ERROR	$A9E9～$A9F4
NO BUFFERS AVAILABLE	$A9F5～$AA08
FILE TYPE MICMATCH	$AA09～$AA1A
PROGRAM TOO LARGE	$AA1B～$AA2B
NOT DIRECT COMMAND	$AA2C～$AA3D

只要把西文ASCⅡ码换成汉字代码就可使输出的信息汉字化。比如把“SYNTAX ERROR”改成“语法错”，可在监控状态下键入：A9E9：7F 52 6E 7F 35 25 7F 32 6C 20 20A0。最后的20 20 A0是“空格”的ASCⅡ码，其中最后一个字节是最高位为“1”的ASCⅡ码。这是每个出错信息的结束码，不能是汉字代码，这一点应该注意。根据以上规律，我们可以用自己喜欢的字来代替出错信息，当然也可以把英文信息转换成小写字母，或用拼音来代替。

把本文给出的数据输完后，用BSAVE CERROR，$A974，L$CB来存盘，在HELLO程序中增加一句BLOAD CERROR，A$A974。当引导DOS后，就可实现上述功能。当然也可INIT一张新磁盘。

A974	7F	4D	32	7F				
A978	56	2E	7F	47	6A	7F	54	7A
A980	7F	49	7C	20	42	41	53	49
A988	43	A0	7F	30	4D	7F	49	7C
A990	7F	32	6C	20	A0	7F	52	4F
A998	7F	4F	32	7F	2F	1F	7F	39
A9A0	20	20	20	A0	7F	4D	5D	7F
A9A8	49	7C	7F	3D	5C	20	A0	7F
A9B0	4D	5D	7F	32	4A	7F	4D	43
A9B8	7F	3B	7D	20	A0	7F	3D	6C
A9C0	7F	38	44	7F	31	75	7F	32
A9C8	6C	20	20	A0	49	4F	7F	31
A9D0	75	7F	32	6C	A0	7F	44	4B
A9D8	7F	41	79	20	20	A0	7F	51
A9E0	50	7F	3B	52	7F	4A	77	20
A9E8	A0	7F	52	6E	7F	35	25	7F
A9F0	32	6C	20	20	A0	7F	4D	43
A9F8	7F	3B	7D	7F	49	7C	7F	31
AA00	29	7F	31	75	20	20	A0	4C
AA08	C5	7F	3F	5F	7F	4F	4C	7F
AA10	30	39	7F	46	77	7F	44	63
AA18	20	A0	C8	7F	43	59	7F	32
AA20	65	7F	51	53	7F	41	79	20
AA28	20	20	A0	C5	7F	31	4B	7F
AA30	4F	71	7F	55	4F	7F	49	37
AA38	7F	52	42	20	20	A0	8D	

苹果机的数据转换

安徽安庆市地震局　　黎　捷

本文主要对该机使用最多的DOS和CP/M两种操作系统的数据转换进行了一些探讨，实现了三种形式的数据转换。

一、浮点BASIC各数据形式间的转换

在浮点BASIC中常用的数据输入方法有三种：

1、直接赋值法：如，行号A=××；

2、DATA语句提供数据：如，行号DATA××，××……等。

3、键盘输入语句给变量赋值：如，行号INPUT变量1，变量2等。

三种输入法相互之间缺少联系。输入后的数据保存形式主要有DATA和文本文件两种。在浮点BASIC中可以利用程序使数据的DATA形式转换成文本文件的形式。而未给出文本文件转换成DATA的形式。这样在一定程度上限制了数据的通用性。

在浮点BASIC中，任何数据(包括DATA语句中的)和程序都可把它们建立成一个文本文件。利用DOS提供的EXEC命令又可以使文本文件以原样返回内存。

利用这一原理，在建立文本文件时就把行号和DATA放在一串数据的前面，存入文本文件。再用EXEC命令把文本文件调入内存。这时内存的数据是以DATA的形式存在。从而实现一般数据→文本文件→DATA语句的转换。

(1)DATA转换成文本文件：

由DATA转换成文本文件，可通过OPEN、READ、WRITE等命令很容易实现。

(2)INPUT转换成DATA形式：

用INPUT给若干变量赋值后，通过OPEN、WRITE建立文本文件，在每组数据前应加上行号和DATA。这样建立的文本文件是一个含有行号、DATA、数据三部分的文本文件。EXEC命令把该文本文件调入内存后。用LIST命令就可以看到用INPUT语句输入的数据就以DATA形式出现了。

这样，即完成了数据由INPUT⇒DATA的转换，也包含了文本文件转换为DATA的过程。(见程序一)。

程序一：

```
10  INPUT "N,M=";N,M
20  DIM X(N,M)
25  C$=CHR$(34)
30  FOR I=1 TO N
40  FOR J=1 TO M
50  PRINT "X(";I;"、"J;
60  INPUT ")=";X(I,J)
70  NEXT J
80  INPUT"Y$=";Y$(I)
90  NEXT I
95  INPUT "FILE NAME";FI$
100  D$=CHR$(4)
110  PRINT D$;"OPEN "FI$""
120  PRINT D$;"WRITE"FI$""
130  FOR I=1TO N
140  K=I+500
150  PRINT K;"DATA";
160  FOR J=1 TO M-1
170  PRINT X(I,J);",";
180  NEXT J
185  PRINT X(I,M);",";
190  PRINT C$;Y$(I);C$
200  NEXT I
210  PRINT D$;"CLOSE"
220  END
```

100——210建立一个含有行号、DATA及数组变量的顺序文本文件。

在程序中直接给字符串变量加双引号运行时会发生错误。

二、DOS和CP/M操作系统间的转换

用CP/M操作系统中的APDOS程序很方便的将DOS操作系统盘片上的数据文件转换到CP/M操作系统盘片上。但是CP/M系统中的数据却不能向DOS系统转换。

DBASE Ⅱ具有很强的管理数据的功能，如能转换成DOS系统中的数据文件，就可为DOS中的浮点BAISC享用。给数据处理增加了一条途径。特别是数据较多时，可避免重复输入工作。也拓宽了数据库软件的交流。

CP/M和DOS虽是两个不同的操作系统，但对数字、字符的内码来说，却是完全相同的。利用这一原理在CP/M中首先用DDT把数据调入内存，再移到内存

高地址处。为避开系统占用的内存空间，最好存在5000H以上的内存区；再进行热启动DOS操作系统。用程序从内存中读出高地址中的数据，然后存到数组变量中，当读完内存后再建立一个文本文件。可根据需要用上节介绍的方法转换成所需的数据形式。

转换的关键在于正确的给出每个变量的长度。在CP/M系统中文本文件在内存中的存放格式有三种：

(1) | 变量1 | 变量2 | … | 回车 | 换行 | 各变量按顺序排列，以回车和换行来结束一条记录。

(2) | 变量1 | ， | 变量2 | ， | … | 回车 | 换行 | 各变量之间均有逗号分开，以回车和换行结束一条记录。

(3) | 变量1 | 回车 | 换行 | 变量2 | 回车 | 换行 | 各个变量都以回车和换行结束。

变量长度的给出有两种方法：

1、在转换程序中逐个在程序中给出，包括每个变量的类型等(见程序三)。

2、在内存中寻找变量间的分隔符(即逗号)或回车换行符的位置，再计算出该变量的长度。

三、汉字的转换

在数据文件的转换中经常有汉字，汉字的转换涉及到汉字的内码问题，各种版本的汉字系统其汉字内码也各不相同。即使用同一汉卡，在不同的系统中的汉字内码也不相同。目前常用的汉字内码有2字节、3字节、4字节等。但无论那种汉字系统，尽管内码各不相同，但它们的区位码却是相同的。因此汉字的转换必须以区位码为中间媒介。

下面以苹果兰汉卡为例对两种类型汉字编码间的转换进行讨论。

1、带有标志符的编码

以汉字的第一个字节为汉字标志码，例如取值127为繁体、126为简体标志，其后跟区、位码。

2、与区位码相对应的编码

它是以汉字内码的前一个或二个字节对应区码，以后一个或二个对应位码来与区位码对应。

苹果兰汉卡的汉字是采用三字节内码。第二字节为区码，取值在48—119间、第三字节为位码，取值在32—126间。在汉字DBASE Ⅱ中.汉字采用二字节内码，第一字节对应区码取值在176—247间，第二字节对应位码取值在160—254间。由此可见这两者的区位对应值仅相差一个系数128。若把DBASE Ⅱ中的汉字转换成DOS系统中的汉字仅需把区位值分别减去128.再在前面加上汉字标志码126或127，结合第二节介绍的方法就可以把DBASE Ⅱ中的汉字转换成DOS系统中BASIC可用的任何一种形式。

程序三是转换一电话簿的例子，该电话簿是在汉字DBASE Ⅱ下建立的，其结构与数据如下：

```
STRUCTURE FOR FILE:  B: DHL. DBF
NUMBER OF RECORDS:  00003
DATE OF LAST UPDATE:  00/00/00
PRIMARY USE DATABASE
FLD    NAME      TYPE  WIDTH   DEC
001    姓名        C     006
002    单位        C     008
003    电话        N     006
* * TOTAL * *                  00021
00001  王小斌  市机械局  513550
00002  张  军  市交通局  543258
00003  李玉海  市化工所  513416
```

现需将其转换成DOS中以DATA形式出现的浮点BASIC中。首先将其考贝成文本文件，再用DDT调入，移至5000H。热启动DOS进入汉字状态，再运行程序三，得到一新文件，然后EXEC新文件。

```
100  DATA  “王小斌” “市机械局” 513550
101  DATA  “张  军” “市交通局” 543258
102  DATA  “李玉海” “市化工所” 513416
```

程序三中，LN表示变量的长度，90句为转换姓名，100句为转换单位，110句用来转换字符与常数，在这里为转换电话号码。390句是用来转换在汉字中夹有的字符或数字。

程序三

```
10   REM 数据间带分隔符逗点，Q$为1，否则为零
20   Q=0
30   HIMEM:24570
40   I1=24575:REM 数据首地址
50   INPUT"N=";N
60   DIM X$(N),Y$(N),Z(N)
70   C$=CHR$(34):REM C$为双引号
80   FOR I=1 TO N
90   LN=6:L$="":GOSUB 380:X$(I)=C$+L$+C$
100  I1=I1+LN+Q:LN=8:L$="":GOSUB 380:Y$(I)
     =C$+L$+C$
110  I1=I1+LL+Q:J=6:A1$="":GOSUB 310:Z(I)=
     VAL (B$)
120  I1=I1+J+2+Q
130  PRINT X$(I),Y$(I),Z(I)
140  NEXT I
200  INPUT "文件名";W$
210  D$=CHR￥(4)
220  PRINT D$;"OPEN"W$""
230  PRINT D$;"WRITE"W$""
240  FOR I=1 TO N
250  K=I+99:REM 行号从100开始
```

APPLESOFT 程序保护探讨

苏州师范专科学校　　赵根林

内容提要　文章系统地阐述了防护 LIST 命令、设计 LIST 效果的原理及其实现方法，同时介绍了破解各种防护的手段。使读者对 APPLESOFT 程序结构有更深入的了解。

APPLESOFT 在接收到命令后首先测试路标单元 $D6 的最高位 Bit7，如果 Bit7＝1 则把所有 BASIC 命令（DOS 命令除外）都看成是 RUN，若 Bit7＝0 则先辨认命令，后转入相应的命令入口执行。假如接收到 LIST 命令，就把 $67 对单元所指地址处的程序码译成 APPLESOFT 语句，显示于屏幕上。所以防止 LIST 可以从三个方面来入手。

一、禁止命令的输入

用 BASIC 命令 POKE214，128 或监控输入 D6：80 可以人为地使零页 $D6 单元的 BIT7＝1，以后不管输入什么 APPLESOFT 命令，均执行 RUN 动作。

要破解这种保护，只要事先设定好 RESET 向量，使强迫进入监控，再把 $D6 的 Bit7＝0。也可以在 DOS 中使用"FP"命令，此命令使 $D6 的 Bit7＝0，但同时清除了程序，所以还必须想办法恢复程序（这可参阅北师大出版社出版的《苹果 Ⅱ BASIC 程序设计》）。

这种保护法不适用于汉卡中文状态，因为汉卡将此单元用于光标定位。

二、修改程序区行讯息

APPLESOFT 程序行讯息包括链指针、行号、语句行实体及结束标志。其中键指针指示了下一行语句所在的位址，而行号一般是依次递增的。如下列的示范程序（下文凡谈及"示范程序"均指该"示范程序"原样，不考虑各项修改；内存内容也是如此）。

```
10 TEXT :HOME:POKE32,7:PRINT
20 PRINT"+………………+"
30 PRINT "! BY ZHAO GENLIN!"
40 PRINT "! ON JAN. 1,1990!"
50 PRINT "! ……………+"
60 POKE 32,0
```

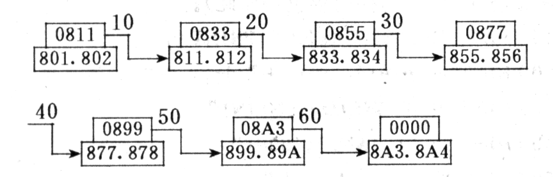
RUN（结果略）程序在内存中的存放格式如下：

```
0800-00 11 08 0A 00 89 3A 97
0908-3A B9 33 32 2C 37 3A BA
0810-00 33 08 14 00 BA 22 2B
0818- 2D 2D 2D 2D 2D 2D 2D 2D
0820- 2D 2D 2D 2D 2D 2D 2D 2D
0828- 2D 2D 2D 2D 2D 2D 2D 2D
0830- 2B 22 00 55 08 1E 00 BA
```

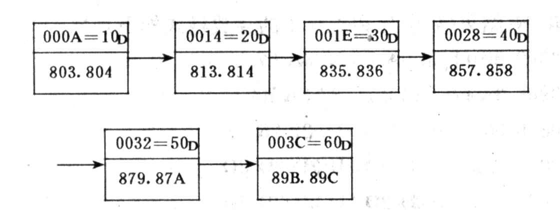
```
260 PRINT K;"DATA";X$(I);",";Y$(I);",";Z
    (I)
270 NEXT I
280 PRINT D$;"CLOSE"W$""
290 END
300 REM 读和转换常数、字符子程序
310 B$="":B2=0
320 FOR K=1 TO J
330 B2=PEEK (I1+K)
340 B$=B$+CHR$(B2)
350 NEXT K
360 RETURN
370 REM 读和转换汉字子程序
380 FOR J2=1 TO LN
390 ZF=PEEK(I1+J2):IF ZF<160 THEN B$=
    CHR$(ZF):L$=L$+B$:GOTO 470
400 H=PEEK(I1+J2)-128
410 IF H<48 OR H>119 THEN 470
420 H2=PEEK(I1+J2+1)-128
430 IF H2<32 OR H2>126 THEN 470
440 H$=CHR$(126)+CHR$(H)+CHR$(H2)
450 L$=L$+H$
460 J2=J2+1
470 NEXT J2
480 RETURN
```

```
0838- 22 21 20 20 20 20 20 42
0840- 59 20 5A 48 41 4F 20 47
0848- 45 4E 4C 49 4E 20 20 20
0850- 20 20 21 22 00 77 08 28
0858- 00 BA 22 21 20 20 20 20
0860- 20 4F 4E 20 4A 41 4E 2E
0868- 31 2C 20 31 39 39 30 20
0870- 20 20 20 20 21 22 00 99
0878- 08 32 00 BA 22 2B 2D 2D
0880- 2D 2D 2D 2D 2D 2D 2D 2D
0888- 2D 2D 2D 2D 2D 2D 2D 2D
0890- 2D 2D 2D 2D 2D 2D 2B 22
0898- 00 A3 08 3C 00 B9 33 32
08A0- 2C 30 00 00 00 01 54
```

链指针如下图(图示格式:下面是链指针存放地址,上面是链指针内容):



行号存放如下图(格式同上):



LIST 子程序就是利用行讯息中的链指针来寻找下一行语句存放地址的,而 **RUN** 不利用链指针。行号只起标识及供 **GOTO**、**GOSUB** 使用,**RUN** 也是不利用行号的。据此,将行讯息作适当的修改,就能得到多种防护措施,而不影响运行结果。不过,**GOTO**、**GOSUB** 需要用到行讯息,所以修改应谨慎。

1、反复列印同一行

如果使本行行讯息的链指针指向本行首地址,则 **LIST** 时,此行程序反复列印,直到按下 **CTRL-C** 或 **RESET**。

作为例子,修改第一行链指针单元使之指向第一行首址 $801。在 **APPLESOFT** 状态下输入 **POKE**2049,1 或监控下输入 801:01,则 **LIST** 使示范程序第一行反复列印,而运行结果不变。

2、只列印第一行

在 **APPLESOFT** 状态下输入 **POKE**2049,0 或在监控下输入 801:0,则 **LIST** 只印出一行,运行结果不变

此修改后,$800、$801 单元内容均为"00",列印时误认为程序结束而不再继续,而 **RUN** 是以连续三个"00"作为结束点,因此不受形响。

3、隐没一些重要的语句行

程序中往往有一些叙述是重要的,我们希望在 **LIST** 时将它们隐藏起来而不影响运行。

如示范程序中,希望把 20~50 号语句隐藏起来,则可把第 10 号语句行的链指针直接指向 60 号语句的首址。在 **APPLESOFT** 状态下输入 **POKE**2049,153 或监控下输入 801:99。

注意,**GOTO**、**GOSUB** 的目标行不得遮没,否则显示 **UNDEF'D STATEMENT ERROR**。

要破解改变链指针的保护,最简单的是重新输入一任意行号,**APPLESOFT** 会把整个程序区的链指针重新接好。

要防止这种破解可人为地破坏程序尾指针 $**AF**、$**B0** 单元,这样使不能输入任何语句,但可 **RUN**,不过此时已不能存盘了。

4、修改行号讯息

行号对 **RUN** 不起作用,可以把它改成各种形式,从而使得程序难以修改。

例一、使行号不依次增大

对示范程序作如下修改:

*803:3C N 813:32 N 835:28

RUN 结果不变

LIST 结果如下:

```
60 TEXT:HOME:POKE 32,7:PRINT
50 PRINT "+………………+"
40 PRINT"! BY ZHAO GENLIN!"
40 PRINT"! ON JAN. 1,1990!"
50 PRINT"+………………+"
60 POKE 32,0
```

这样的修改对 **GOTO**、**GOSUB** 等转向语句也须注意,此种语句搜寻目标行每次都是从程序首址开始,利用行讯息依次向下找,找到后执行,如果发现目前行号已大于目标行号,但仍未找到,则显示 **UNDEF'D STATEMENT ERROR**。

例二、大行号保护

APPLESOFT 规定语句行号最大不超过 63999,如果输入大于限定行号的语句、显示 **SYNTAX ERROR**。因此可以修改行号建立大于限定行号的语句行而不影响运行结果。

对示范程序作如下修改:

*803:3C N 813:FF FF N 835:FE FF N 857:FF FF N 879:FF FF

LIST 结果如下:

```
60 TEXT:HOME:POKE 32,7:PRINT
65535 PRINT"+………………+"
65534 PRINT"! ON JAN. 1,1990 !"
60 POKE 32,0
RUN
```

如果GOTO、GOSUB等转向语句目标行号改为大行号，运行时显示SYNTAX ERROR。

破解这种保护只能再修改内存。

从以上的例子我们还可得到启发，因为转向语句每次搜寻都得从头开始，所以在较长程序中把子程序集中放在程序的开头，就可减少搜寻时间，从而加快了运行速度。

三、装饰LIST效果

LIST输出需要经过监控的COUT程序，在COUT中字符先与$32单元内容作AND操作，所以$32单元的值不同，输出也不同。另外，控制字符是不能在屏幕上看到的，它在COUT时只产生一些特殊作用，如退格、发声、换行等等，所以加入一些控制字符可装饰LIST效果。

1、修改显示方式代码单元

$32这个单元的值决定了显示方式，$FF为正常，$7F为闪烁，$3F为反相。

如果改变其值，则输出结果变得非常奇怪，如改为$80，则输出信息全为空格。

破解它的最简单办法是按RESET键，或在监控下把它的值改为$FF。

2、CTRL-H的利用

CTRL-H即为删除键"←"，它使输出光标后退一格。利用它可将程序隐没及伪装。由于这个字符无法直接输入到程序中，必须采用先预留空间，后直接填入的方法。

例一、隐没程序

上文提到过利用修改链指针可使LIST只列印一行，但还是列出了一行。如何一行也不列出呢？可利用CTRL-H。

方法如下：(以示范程序为例)

①加入第5号语句：5REMX(注意REM与X间不留空格)。

②列印第5号语句(REM与X间自动插入一空格)：

```
5REMX
LIST5
5 REM X
```

③自X往前数(X本身不算)，凡是可见字符都算数，数到5为止，共有七个字符空间。

④重新键入第5号语句：

```
5REMXXXXXXX
```

REM后七个"X"，七个空格。

⑤进入监控：

```
0800-00 15 08 05 00 B2 58 58
0808-58 58 58 58 58 20 20 20
0810-20 20 20 20 00
```

自$806到$80C的七个"58"即是"X"的ASCII码，之后的七个"20"是空格的ASCII码。

⑥将$806～$80C的七个"58"改为"08"，"08"即为CTRL-H的ASCII码。

⑦回到"]"，LIST时，5号语句不见了，接着POKE2049,0修改链指针，整个程序都不见了。

例二、伪装程序

现在把示范程序的60号语句伪装成POKE64，10，方法如下：

仿照例一的①～④步，计算得到X与3间的字符空间为10个，重输60号语句如下：

```
60 POKE 32, 0: REMXXXXXXXXXX64, 10: REM KEYWORD
```

进入监控可看到$8A4～$8AD的十个"58"，将它改为"08"。回到"]"，LIST结果：

```
60 POKE 64,10: REM KEYWORD
```

破解CTRL-H保护的最有效方法是用速度函数SPEED来降低输出速度，使字符一个个输出，当在某一行停较长时间，且过一会前面已显的字符在发生变化，则可断定陷阱就在此处。

3、CTRL-D的利用

程序中使用DOS命令，第一个字符必须为CTRL-D。这是因为DOS视讯中断处理子程序在做COUT前会判断回车后第一个字符是否为CTRL-D，若是则收集CTRL-D后的DOS命令，继而跳到DOS命令剖析子程序执行。利用这一原理，在程序中埋设DOS命令，使LIST时也产生DOS动作，此法一定要DOS在内存时方有效。且应注意，命令必须埋在RUN不到的地方，如REM、DATA、THEN、GOTO、GOSUB等描述项后。又因DOS视讯中断程序认可DOS命令一定要是回车后第一个字母为CTRL-D，所以在加入CTRL-D和DOS命令前必须先产生回车"RETURN"。

由于回车字符无法直接加在程序中，仍采用先留空间后填入的办法。RETURN和CTRL-D的ASCII码分别为"0D"和"04"。

以在示范程序中加入DOS命令"INT"为例来说明实现方法。

加入语句5REMXXINT，进入监控将$806、$807单元的"58"改为"0D"和"04"，回到"]"状态，LIST结果：

对《APPLE 汇编语言高速排序》一文的异议及改进

贵刊 1989 年第 6 期中《APPLE 汇编语言高速排序》一文所附汇编语言子程序，要改正 4 处方可正常运行：

	误	正
374	A2	02
39A	B4	D4
39B	B0	D0
3C2	AB	A8

在文中也有一处错误："当 a=0，b=数组长度时，即是对整个数组的全部数据进行排序。"这句话与前述"子程序对从 a+1 起的 b 个数据进行排序……"是相矛盾的。事实上，数组中头一个下标为 0 的变量不能参加排序。如要对全部数据排序需作如下调整：

```
*335:EA
*39E:EB
*3A0:1A 90
*3A5:B0
```

这样改后，成为对从 a 至 a+b 共 b+1 个数据进行排序。但这只是修正，算不上改进。

笔者为排序达到全部数据，作了尝试性的改进，改后的程序附后，用法与原文程序完全一样，而且可以排 N+1 个，即数组的全部数据。该程序长度比原文程序少了一个字节，但速度显著提高，对 N=1000 的数组排序仅需 5 秒。

由于该程序中几处使用了未公布的 6502 怪指令码，可读性较差，并且仅适于 6502，对于 65C02 之类处理器的微机还要作些相应修改。

原文程序和本文程序均仅限于实型数组的排序，不能用于整型或字符串的数组。

```
0300-  A9  0B  8D  F6  03  A9  03  8D
0308-  F7  03  60  20  E3  DF  84  3D
0310-  85  3C  20  BE  DE  20  67  DD
0318-  20  52  E7  A5  50  85  3E  85
0820-  40  A5  51  85  3F  85  41  A7
0328-  41  87  3A  87  3B  A5  40  46
0330-  41  6A  A8  05  41  F0  D3  98
0338-  85  40  0A  0A  86  FB  26  FB
0340-  65  40  85  FA  A5  FB  65  41
0348-  85  FB  A5  3E  38  E5  40  85
0350-  42  A5  3F  E5  41  85  43  A5
0358-  3B  A4  3A  85  4E  84  4F  85
0360-  FF  98  0A  26  FF  0A  26  FF
0368-  65  4F  AA  A5  FF  65  4E  A8
0370-  8A  65  3C  AA  98  65  3D  A8
0378-  8A  20  F9  EA  18  A5  5E  65
0380-  FA  85  60  A5  5F  65  FB  85
0388-  61  20  B6  EB  B0  12  E6  3A
0390-  D0  02  E6  3B  A5  42  C5  3A
0398-  A5  43  E5  3B  90  89  B0  B7
03A0-  A0  04  B3  5E  B1  60  91  5E
03A8   8A  91  60  88  10  F4  A5  4F
03B0-  E5  40  A8  A5  4E  E5  41  90
03B8-  D5  D0  A0  A5  4F  C5  40  90
03C0-  CD  A9  00  B0  96
```

唐山白玉瓷厂　　张亭

5 REM

LANGUAGE NOT AVAILABLE

因为此时无 INTEGER BASIC 在内存，否则将进入整数 BASIC 状态。

所有 DOS 命令均可埋设，如果埋设的命令错误或不是 DOS 命令，则 LIST 时显示 SYNTAX ERROR。不过得提醒一下，即使命令正确，在完成 DOS 动作且把剩余程序 LIST 完后，均会显示 SYNTAX ERROR，除非执行 DOS 命令后，进行内存初始化。如"FP"、"PR＃6"、"LOAD 文件名"等。常用的是 FP 和 PR＃6，最好的办法是用"INIT"命令。

4、其他控制字符的利用

还有一些控制字符可用来装饰 LIST，这里只作一简单说明，具体可仿以上几个的运用方法。

CTRL-T 反白及回复，必须配对使用，使夹在其中的字符反白。但在一些兼容机上只有接通打印机才反白。

CTRL-J 空行，此字符无法直接输入。

CTRL-G 发声。

如果为防止他人把程序 LIST 到打印机上，则可选用的控制字符有：

CTRL-Q 进行高分辨页的硬拷贝；CTRL-L 产生换页功能；CTRL-N 字体放大，碰到换行自动取消；CTRL-O 字符缩小；CTRL-R 取消字体缩小；等等。更可以使用 ESC 码来产生更多的打印效果，具体请参阅有关打印机使用手册。

·苹果园地·

利用微机绘制RLC电压谐振曲线

黑龙江呼兰师专　吴全有

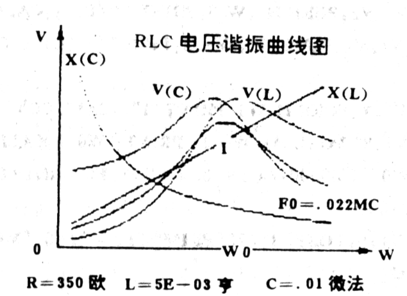
本程序利用微机绘图方法对电阻(R)、电感(L)、电容(C)串联回路的电压,画出了电感上的电压V(L)、电容上的电压V(C)和感抗X(L)、容抗X(C)及电流I随角频率W的变化曲线,并标示出谐振点的角频率W0,计算出谐振频率F0。可供电气工作者计算不同LC值的谐振频率。可以给Q不同值,而看其在不同品质因数(Q值)下曲线的变化规律。在LC值不变的情况下,当Q值大(R较小)时,V(C)、V(L)及I的尖峰较大;当Q值小(R较大)时,V(C)、V(L)及I较平坦。因此,该程序还可做为研究人员研究电压谐振的助手,也可做为教学人员实验和演示之用。

为使图中能绘出谐振点,在程序的800~850行设置了可变参量K,借以改变横座标W的起止取值范围。同时为了尽可能扩大RLC的取值范围,程序的250~280行设置了比例系数,保证了各条曲线不超出图形的边界。本程序在APPLE Ⅱ机调试通过。

使用方法:

1、开机自动进入中文状态,键盘输入R、L、C值,回车后开始绘图。

2、绘图完成后,敲空格键即自动联机打印图形。敲其它键,则仃止运行。




```
10 HIMEM:36864: POKE 37984,0: PR # 3:PRINT"  ":
   POKE 43603,3,POKE 43604,192+3:POKE 43605,48:
   POKE 43606, 192+3:POKE 54,189, POKE 55, 158:
   POKE 56,129:POKE 57,158:FOR I=1 TO 15:GET A$:
   NEXT
20 D$=CHR$(4):HM=49941:CC=49944:H1=214:V1
   =215
30 K=600:V=50
40 CALL HM
50 POKE V1,2:POKE H1,6: PRINT"绘制RLC电压谐振曲线
   图":CALL CC
60 POKE V1,5:PRINT "说明:":CALL CC
70 POKE H1,9:PRINT "电感单位(亨利)":CALL CC
80 POKE H1,9: PRINT "电感单位(微法)":CALL CC
90 POKE H1,9:PRINT "电阻单位(欧姆)":CALL CC
100 INPUT "L=";L
110 INPUT"C=";C0
115 INPUT "R=";R
120 CALL HM
130 C=C0*0.000001
140 HGR2:HCOLOR=3
150 HPLOT 20,5 TO 20,150 TO 250,150
160 HPLOT 20,5 TO 18,14
170 HPLOT 20,5 TO 22,14
180 HPLOT 250,150 TO 241, 148
190 HPLOT 250,150 TO 241,152
200 POKE V1,0:PRINT "V":CALL CC
210 POKE V1,8: PRINT "0";: CALL CC
220 POKE V1,8: POKE H1,32: PRINT "W": CALL CC
230 W0=1/SQR (L*C):W0=INT(W0*100)/100
235 GOSUB 800
240 I1=V/(SQR(R↑2+(W0*L-1/(W0*C))↑2))
250 K0=I1*R/80
260 K1=I1*W0*L*1.3/120
270 K2=1/(30*K*C)/130
280 K3=(220*K*L)/110
285 W1=W0/K
290 HPLOT W1,150 TO W1,147
300 FOR W=30 TO 220
305 P=W*K
310 I=V/(SQR(R↑2+(P*L-1/(P*C))↑2))
330 Y=147-I*R/K0
340 HPLOT W,Y
350 VL=I*P*L/K1: V2=147-VL
390 HPLOT W,V2
400 VC=I/(P*C)/K1
410 V2=147-VC
```

CATALOG 功能扩充

广州三十二中　　谢敬辉

《苹果园》八九年第四期的《对 CATALOG 命令的改进》提出了使 CATALOG 具有与 DIR 相似的功能，但只是改进了列目录的范围。笔者为进一步扩充 CATALOG 的功能，编写了以下程序，使它在 CATALOG 时列出卷号及空扇区数，还列出已删的文件名，名前带“#”。

先键入本程序，再 300G 便可具有以上功能。$300—$31F 为更改 CATALOG 部分入口，$320—$365 为列出空扇区数，$366—$36D 为对已删文件名的处理。

由于本程序与《对 CATALOG 命令的改进》一文所提方法并无冲突，互不干涉，所以两种方法可同时并用，只须修改部分入口地址即可。

另外，还有几个小技巧可使 CATALOG 功能得以扩充。

1. 将 $DA 存入单元 $AE1C，在列目录时可列出文件名的 ASCII 码值，起到对加怪符保密文件名的破密和使用中文名称的文件的作用，只须以 ASCII 码值存入 $AA75 文件名缓冲区，再调相应功能程序，便可使用到那些不易看清的文件。复原只须 AEIC：ED 即可。

2. 由于在中文下屏幕行数的不同，所以更改每帧显示文件名个数是有必要的。可更改 AE3D 和 ADA4 两单元内容，原值分别为 $15(个数)和 $16(个数加1)

3. 若列长目录时去掉暂停功能可 AE34：60，复原只须 AE34：CE。

4. 有时不须列完全部目录时，中途退出 CATALOG 可按 CTRL—C 或 QUIT 键，修改如下：AE39：4C 69 BA。BA69：20 0C FD C9 83 F0 03 4C 3C AE 4C 7F B3。若要复原可 AE39：20 0C FD。

说明一点：由于系统的 DELETE 动作，列已删文件名时，第三十个字符变为一些反相的混乱字符，是没有什么意义的，并不影响该功能。以上全部修改及程序均在 CEC—II 及 APPLE 机上通过。

```
0300-  A9  4C  8D  C3  AD  8D  C6  AD
0308-  A9  03  8D  C5  AD  8D  C8  AD
0310-  A9  20  8D  C4  AD  A9  66  8D
0318-  C7  AD  A9  EB  8D  DA  AD  60
0320-  20  2F  AE  A9  59  A0  03  20
0328-  3A  DB  A9  00  85  40  85  41
0330-  A0  8C  18  B9  F2  B3  F0  0E
0338-  0A  90  FB  48  E6  40  D0  02
0340-  E6  41  68  18  90  F0  88  D0
0348-  E9  A6  40  A5  41  20  24  ED
0350-  20  2F  AE  20  2F  AE  4C  C9
0358-  AD  C6  D2  C5  C5  A0  D3  C5
0360-  C3  D4  CF  D2  A0  00  A0  A3
0368-  BD  C8  B4  4C  E4  AD  00
```

```
420 HPLOT W,V2
480 XC=(1/(P*C))/K2:XL=P*L/K3
500 IF 150 -XC>0 THEN HPLOT W,150-XC
510 HPLOT W,150-XL
520 NEXT W
530 V2=(150-XL)/20
540 POKE V1,V2:POKE H1,27:PRINT "X(L)":CALL CC
550 POKE V1,1:POKE H1,3:PRINT "X(C)":CALL CC
560 POKE V1,9: PRINT "R=";R;欧 " ";
565 PRINT "L=";L;"亨 "
570 PRINT "C=";C0;"微法";:CALL CC
580 POKE V1,0: POKE H1,9:PRINT"RLC 电压谐振曲线图":CALL CC
590 GOSUB 700
600 GET G$
610 IF G$<>" "THEN E
620 PR# 1:POKE 1913,2
630 PRINT CHR$(17)
635 PR# 0
640 E
700 V2=(147-I1*W0*L/K1)/20
703 Y=(147-I1*R/K0)/20+1
705 W=W1/8
710 POKE V1,V2:POKE H1,W-6:PRINT "V(C)";:CALL CC
720 POKE V1,V2:POKE H1,W+2:PRINT "V(L)";:CALL CC
730 POKE V1,Y:POKE H,W1:PRINT "I";:CALL CC
740 POKE V1,8: POKE H1,W+1: PRINT "W0";:CALL CC
745 F0=W0*.000001/(2*3.14153):F0=INT(F0*1000)/1000
750 POKE V1,6:POKE H1,23:PRINT "F0=";F0;"MC";:CALL CC
755 POKE V1,8:CALL CC
760 RETURN
800 FOR W=1 TO 100
810 IF W0/K>150 THEN K=K+100:GOTO 840
820 IF W0/K<50 THEN K=K-100:GOTO 840
830 GOTO 850
840 NEXT
850 RETURN
```

计算机辅助教学实验研究初探

——计算机辅助教学效果分析之二

烟台师范学院 李克信 北京师范大学 陈琦

一、问题的提出

计算机辅助教学(CAI)起源于六十年代初期。当时,行为主义心理学家斯金纳提倡的程序教学和教学机器运动风行于欧美各国,人们把这一运动作为当时教学改革的一个重要组成部分并寄予极高的期望。由于电子计算机的速度快、容量大、灵活以及具有人机对话等功能,可以弥补机械装置的教学机器的一些不足,自然就成为比较理想的自动化教学机器①。早期的CAI实际上就是一种自动化程度较高的程序教学。

六十年代至七十年代初,在美国研究CAI最负盛名的项目有:(1)斯坦福大学的P.萨佩斯和R.阿特金森从1963年起开始的CAI的研究和开发工作。最初是对小学生和能力低下的学生用CAI作数学和语言学习的操练和练习,后来又发展到利用CAI作个别指导。他们的研究旨在搞清作为小学课程的组成部分、在一段校长的时期内使用个别化指导的CAI来讲授数学和阅读课是否适当。结果表明,在阅读教学方面,实验前学生在接受了10个月的正规课堂指导后,平均阅读能力才达到5.4个月的水平,而使用计算机教学后,学生仅接受8个月的指导,其平均阅读能力达到9个月的水平②。(2)伊利诺埃大学开发的PLATO系统(为Programmed Logic for Automated Teaching Operation的缩写,意即"用于自动教学操作的程序逻辑")。1959年首先创建了一次只能处理一个终端的PLATO—I系统;1968年开发了大规模的PLATO—Ⅳ系统,能连接分布于800哩范围内的4000个学生终端,同时教几百种课程③。研究结果表明,许多学生和教师改变了对PLATO材料的态度。在数学学习方面,PLATO的CAI材料明显优于传统的课堂教学计划Alderman,1977)④。(3)迈特公司和杨伯翰大学新开发的TICCIT系统⑤。其特点是将几个独立的、由计算机控制其终端的分散系统连结起来,目的是利用小型计算机和电视技术向公共学院学生提供英语、数学的CAI课程和教学程序。研究结果表明,该CAI教学效果优于传统的正规课堂教学。

六十年代后期,由于行为主义理论本身的缺陷,大量粗制滥造的程序课本和教学机器涌入市场,致使程序教学受到种种批评。而对CAI作用的过分夸张宣传(特别是宣传它能代替教师的作用而遭致许多教师的抵制),机器价格的昂贵,适用软件又不多,也使CAI的声誉大降,一个时期内几乎销声匿迹。七十年代中、后期,由于计算机技术的飞速发展,机器价格的大幅度下降,特别是微型计算机的开发投放市场,才使学校中利用计算机辅助教学的运动再度掀起高潮。

计算机辅助教学的效果究竟如何?廿多年来众说纷纭,莫衷一是,实验研究的结论也一直存有争议。最初是充分肯定,例如前面新举的一些持肯定态度的研究。随后,随着对程序教学运动的种种非议,对CAI热情的下降,出现一些持否定结论的研究。例如,伯少尔德和撒克斯Berthold and Sachs,1974)在研究中发现教师的教学优于计算机辅助教学。研究对象是六名情绪障碍儿童,对他们进行乘法和拼写等方面的教学,采用了三种不同形式的教学:计算机教学、教师教学和两者混合教学。结果是教师的教学效果优于其它两种形式⑥。1978年蔡和波尔(Tsai and Pohlì)也作了CAI与教师教学的比较。他们把45名大学生分为三组(CAI教学,教师讲授和两者混合),用四种不同的方式测验学生计算机程序设计的成绩(小测验、家庭作业评等、学期设计等级和最后测验分数)。从家庭作业和学期设计等级的数据表明,各实验组之间无显著差异⑦。八十年代以来,一些研究工作者对大量的CAI研究进行了元分析*后,又趋于持积极的肯定态度。伯恩和鲍兹曼(Burn and Bozeman,1981)通过对一至十二年级学生的数学成绩和CAI教学进行元分析得出结论是:①用CAI补充的数学课在促进学生的成绩方面十分有效;

* 元分析是一种新的统计方法,对同一课题的大量研究报告作统计分析从而得出一种倾向性的结论——作者注。

②CAI 的操练和练习对成绩高和能力差的学生在各级水平上都比较有效，而中等水平的学生的成绩没有显著的提高；③没有证据表明这些结果是由于实验的特征而产生的[8]。还有人在运用元分析时发现，在 28 个 CAI 的教学实验研究中有 23 个实验报告表明 CAI 与传统教学成绩存在显著差异，这 28 个研究中学生成绩 CAI 平均高出 0.47 个标准差。

总之，国外廿多年的 CAI 研究中，CAI 与传统课堂教学相比，其效果仍难定论。结论还需进一步证实。

我国有计划地开展计算机教育试验是从 1982 年开始的。由于受到"程序设计是第二文化"这一思想的影响，中学计算机教育的试验开始就是开设以程序设计语言为主的计算机选修课。直至 1986 年第三次全国中学计算机教育工作会议上才把计算机辅助教学提到工作日程上，在教委的工作报告中把"组织力量开发计算机教育软件，在有条件的地区和学校逐步开展计算机辅助教育……"作为安排中学计算机教育工作的三个层次之一。此后，在国家科委、国家计委、国家教委、电子工业部和中国科协五个部委的关怀和支持下，成立了五个部委的协调小组，组织计算机专家研制和开发适合于中国青少年计算机教育的国产化微型计算机——中华学习机，并在七五计划中拨专款开发教学软件。建立了教育软件登录和评审机构。三年来，已先后进行了三次全国性的教育软件评审会。由于我国教育软件的研制开发起步晚，基础薄弱，至今为止，合格的软件还不多，特别是可供教师用作辅助教学的软件更少；因此，在我国利用计算机辅助教学无论从理论上还是从实践上都有许多问题需要得到解决。诸如，在我国的教育制度和教学条件下，计算机辅助教学可能起什么作用？怎样才能起作用？CAI 与教师的教学应该是一种什么关系？在应用 CAI 过程中如何促进学生的基本知识和基本技能的训练？

如同任何一种教学方式一样，计算机辅助教学这种教学方式的效果，除了充分发挥它自身固有的优势以外，还要受到教师的因素、学生的因素、教学软件的因素等影响。探讨这些因素，搞清影响 CAI 效果的条件，对于指导计算机辅助教育的发展无疑是有益的，也是计算机教育研究工作义不容辞的责任。

二. 研究目的

本研究拟探讨以下几个问题：

1. CAI 与传统教学方式效果比较。
2. 学生学习能力不同是否影响 CAI 的效果。
3. CAI 教学中学生的学习兴趣与态度。
4. CAI 教学对学生认知能力的作用。

三. 方法与步骤

1. 被试

本实验采用自然实验法。在课堂正规教学的基础上，利用微机进行辅助教学。被试选自北京 161 中学高中一年级(1)(2)两班，每班人数为 40 人，将每班随机分为实验组和控制组各 20 名，男女人数相等。在数学课上进行实验。

补充实验在北京 161 中学选取初三一个班，44 名学生作为被试。根据英语基础成绩用配对法，随机分为实验组和控制组，实验组在机房上机练习，控制组在教室中学习，练习作业相同。练习时间亦相同。

2. 实验设备和教学软件。

本实验所用微机是国产 Apple—Ⅱ，兼容机 BMC 和 Korin 共 20 台、磁盘驱动器 20 台。

教学所用的软件有国外引进的三个：①美国 Sunburst 公司的"解释图象"(Interpreting Graphic，已汉化)；②美国 Educational Pullishers 公司的"超级绘图"(Superplot；③美国 Sunburst 公司的"方程绘图(Graphing Eguation)。这三个软件中主要内容与我国数学课本中有关内容相配，并经国外专家推荐为优秀软件。此外，还根据教材需要由计算机教师移植和编制了"函数与图象"软件。

在补充实验中采用了国内教师编写的英语介词练习，英语任课教师根据教学需要提供了练习内容。

3. 实验教学内容

实验涉及的内容为高中课本《代数》第一册(甲种本)中的第一章幂函数、指数函数和对数函数中的第三节幂函数和第四节指数函数和对数函数。此外还涉及全日制十年课本中，初中函数中的二次函数。

4. 实验步骤分五个阶段

第一阶段：对高一两班所有被试进行数学摸底测验，以此为基础，将各班一分为二，随机确定为实验组和控制组。此外，还对所有被试进行了认知推理能力测验和隐藏图形测验，以确保实验处理前两组被试水平基本相同。

第二阶段：根据第一阶段进行的分组，对实验组进行实验处理，对控制组进行正常教学，时间从 87 年 9 月起大约持续两个月左右。

在整个实验期间，实验组和控制组基本上都同时在教室中听课。在讲到上述实验内容如幂函数单元时，实验组到机房上课，结合讲解用计算机进行演示。课后实验组上机对所学习的内容进行练习，而控制组按传统方式作书面材料的练习。练习时间都大体相等，内容大体一致。实验班每人累计上机练习 10 课时左右。

第三阶段：在上一阶段教学处理后，对高一(1)(2)两班全体进行后测，考察实验效果。同时，再一次对该两班进行隐藏图形和认知推理能力测验，以考验函数数学对认知能力的效果。

第四阶段，进行学习内容的保持和迁移测验，进一步检查学生学习的效果。此外还进行了有关CAI教学的问卷调查和个别谈话，以补充测验成绩以外的情况。

第五阶段，补充实验，根据实验的结果，于88年3—4月进行了CAI的补充实验。在初中三年级一个班作英语介词练习的实验，以进一步探索影响CAI效果的条件。

四. 实验结果及分析

(一)各实验组和控制组数学知识初始状态

表1. 实验组和控制组数学摸底成绩的方差分析

班级	变异来源	平方和 ss	自由度 df	均方 ms	F值
高一(1)	组间变异	41.86	1	4.86	0.79
	组内变异	1957.86	38	51.52	
	总变异	1999.72	39		
高一(2)	组间变异	9	1	9	0.09
	组内变异	3731.97	38	98.21	
	总变异	3740.97	39		

$F_{0.05}(1.38)=4.10$

$\alpha>0.05$

表1的分析表明，两班实验组与控制组的数学摸底成绩均无显著差异，可以说两班的实验组与控制组初始水平相同。

(二)各实验组和控制组的认知推理能力，识别镶嵌图形能力的初始水平。经方差分析均表明差异不显著，即起点相同。(方差分析表从略)

(三)实验处理对被试认知能力的影响。

表2. 两班推理能力和识别镶嵌图形能力的差异

班级	测验		平均值	标准差	人数	t值	p值
高一(1)	推理测验	实验组	11.5	1.73	20	-0.6	P>0.05
		控制组	11.55	3.24	20		
	镶嵌图形测验	实验组	12.16	5.27	19	0.04	P>0.05
		控制组	12.1	4.69	20		
高一(2)	推理测验	实验组	12.95	1.73	20	0.22	P>0.05
		控制组	12.76	3.48	20		
	镶嵌图形测验	实验组	6.17	4.99	18	0.82	P>0.05
		控制组	4.95	4.21	20		

表2表明两班的实验组在实验处理后对认知推理能力和识别镶嵌图形能力方面没有显著影响。实验组和控制组在这两个测验中的差异不显著。

(四)实验处理对实验组数学学习的影响。

表3. 实验前后两班实验组与控制组数学成绩比较

班级	被试		平均值	标准差	人数	t值	p值
高一(1)	前测	实验组	86.96	11.49	20	0.95	P>0.05
		控制组	89.7	13.46	20		
	后测	实验组	80.85	9.49	20	0.31	P>0.05
		控制组	79.8	11.48	20		
高一(2)	前测	实验组	81.4	16.06	20	0.69	P>0.05
		控制组	81.06	16.62	20		
	后测	实验组	80.35	7.26	18	2.18*	P<0.05
		控制组	73.56	11.69	20		$P_{0.05}=2.02$

表4. 两班实验组与控制组数学学习保持和迁移效果比较

班级			平均值	标准差	人数	t值	p值
高一(1)	保持	实验组	84.89	8.48	19	-0.78	P>0.05
		控制组	86.79	6.1	17		
	迁移	实验组	87.79	9.26	19	0.43	P>0.05
		控制组	86.12	13.19	17		
高一(2)	保持	实验组	75.95	8.5	18	0.66	P>0.05
		控制组	73.73	12.33	20		
	迁移	实验组	80.6	8.34	18	0.81	P>0.05
		控制组	77.4	15.53	20		

表3表明CAI对实验组学生学习数学成绩有所提高，其中高一(1)班实验组略有提高，但从t检验看，差异不显著。而高一(2)班则实验组平均成绩显著高于控制组，t值为2.18，P<0.05。

从表4可见，两班的实验班和控制班在学习的保持和迁移方面成绩差异不显著。但是从总体上看，特别是在高一(2)班，实验组的平均分数略高于控制组。

(五)利用计算机辅助教学对学生学习兴趣和学习态度的影响。

在实验处理后，我们用问卷的方式，向被试提出一些问题，以了解CAI对学生学习兴趣与数学学习态度的影响。其结果见图1，图2。

图1是学生对五个里克特量表式的多重选择题所作的反应。这五个题是：

1. 你是否喜欢用计算机来学习数学？
2. 如果有语文、物理等其它软件，你是否愿意用？
3. 计算机结合函数教学，图形对知识的表示是否形象？
4. 每当上机时，你的感受。
5. 用计算机对理解、记忆函数性质是否有促进作用？

要求学生对每个问题在五个等级中选择一个。如①很喜欢(很愿意、很形象、很愉快、很有促进作用)；②喜欢(愿意、形象……)；③无所谓(一般、无作用)；④不喜欢(不愿意……)；⑤讨厌(很不愿意……)。

在图1中我们把肯定性的①②等合在一起统计，而把选择无所谓或否定的③④⑤等放在一起统计。五个题中选择①②等的学生百分数大大高于选择③④⑤

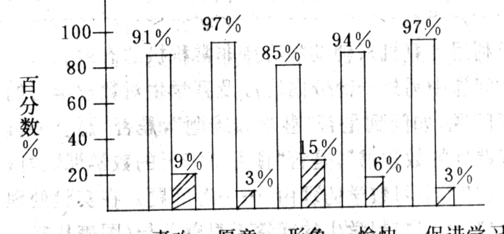


图1 两个实验组被试对用计算机辅助学习的兴趣

肯定 无所谓或否定

等的学生百分数。可见实验组学生在用过CAI之后，

·辅助教学与研究·

中小学CAI软件设计的几点看法

四川宜宾地区教学仪器站 李志刚 宜宾四中 聂运洁

一、引言

计算机辅助教学(简称CAI)主要对象是学生,其核心部分是教学软件。高质量的教学软件可以使学生在丰富多采引人入胜的愉快气氛中主动地学到了知识。能调动学生学习的积极性,获得良好的教学效果。本文就中小学CAI软件设计中的一些问题作一探讨,旨在抛砖引玉。

二、软件设计人员

当前,在广大从事CAI软件开发工作者的努力下,我国中小学CAI软件在数量和质量上都有较大提高。但由于这些软件的设计者大多数是电脑技术人员,缺乏对中小学教材、大纲以及中小学生实际的了解,设计的一些软件教育目标不够明确,往往显得呆板,没有充分发挥计算机作为新型教育手段的优势。因此,对学生吸引力不大。我们认为一个高质量的CAI软件是集教育学、心理学专家以及中小学教师、计算机专业人员之精华于一体的产物。

为了提高中小学CAI软件的质量,设计人员需具备以下知识结构:

1),计算机专业人员的有关知识及实践能力。

2),丰富的中小学教学经验。

3),熟悉中小学教材及大纲。

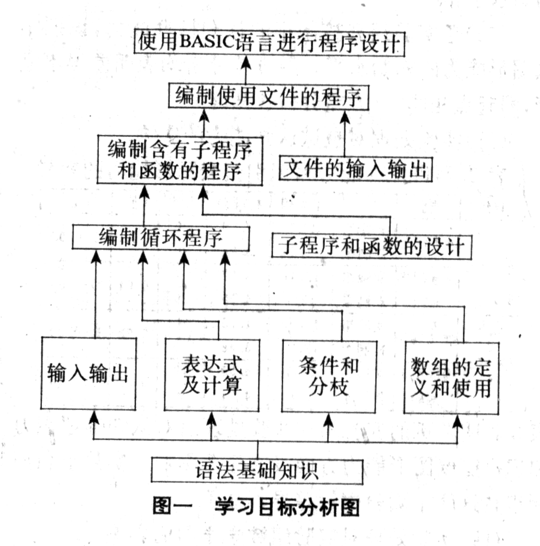
4),具备教育学、心理学及教学法等有关知识。

因此,笔者建议在有条件的地方,组织有丰富教学经验的中小学教师、科技辅导员、软件人员、教育心理学家等参加的开发队伍,发挥各自所长,以期得到一批高质量的CAI软件。

三、教学需求分析

设计CAI软件,必须进行教学需求分析,明确教学对象与教学要求,尤其是对教学要求必须加以分析和细化,使得总的教学目标进一步划分为若干个子目标,同时论证这些子目标存在的必要性和合理性,并画出学习目标分析图。

以开发"BASIC语言"的软件为例。假设总的教学要求是使学生能够用BASIC语言编制程序,选择对话型的CAI模式,则可画出学习目标分析图(图一)。从图中可以看出,最终目标是通过一个个子目标达到的,而每一个子目标的实现,需要学生掌握一系列的概念和技巧,因此,还需要把子目标再进一步细化,即画出子目标中所含的各种概念及其相互关系图,即所谓的概念图(有的CAI可省略这一步)。例如,子目标"BASIC的语法基础"中所含的概念及其关系如图二。



图一 学习目标分析图

对于利用计算机来辅助学习持非常积极的态度。

问卷中另外一部分内容涉及到学生对数学学习的态度问题,如问到是否"喜欢做难题",是否"愿意阅读一些课外的数学书",是否"预习下一天的数学课"、"做课外题"、"学习数学的理由"……等问题。在实验处理后,实验组与控制学生的回答无显著差异(图表从略),说明CAI的使用对学生数学学习态度无显著影响。

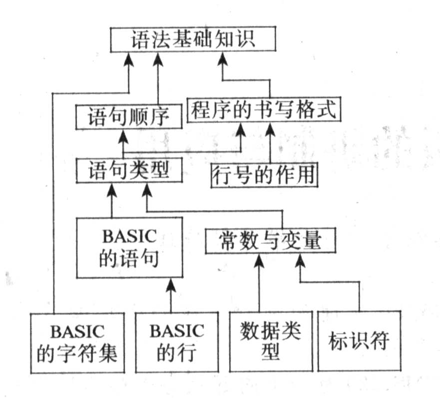
(六)补充实验——英语介词练习的结果分析。

表5.实验组与控制组英语成绩比较

	被试	平均值	标准差	人数	t值	P值
前测	实验组	83.82	10.93	22	0.18	P>0.05
	控制组	83.2	11.7	22		
后测	实验组	78.3	15.54	22	1.66	P>0.05
	控制组	71.82	14.31	22		P.05=1.68

表5中P值虽然大于0.05,也即在0.05的水平上无显著差异,然而t值1.66是接近P0.05=1.68的水平,从平均值看,实验组高于控制组6.48分,应该说CAI对初三学生学习英语介词的成绩是比用传统教学的控制组要好。可以说在P=0.06的水平上,差异显著。(tì(0.06)=1.65,P<0.06)

(未完待续)



图二　概念图

四、体现计算机的特点，充分调动学生的积极性

一个高质量的 CAI 软件不是课本内容的叙述，要体现计算机的特点，有利于激发学生学习的兴趣和主动性、积极性，有利于培养学生的能力，能取得其它教学手段无法取得的效果。因此，要充分利用计算机的绘图功能及音乐功能，通过巧妙结合，达到有声有色的动画效果。如我们在设计初中化学离子化合物的形成这一 CAI 软件时，不只是叙述方程式：

$$Na+Cl \rightarrow NaCl$$

而是通过计算机高分辨作图，描绘出钠原子失电子（用×表示，红颜色绘出），氯原子得到电子（其核外电子用·表示，绿颜色绘出）的动态过程，同时伴有音乐提示。这样使学生在愉快的气氛中学到了知识。体现了计算机作为智能工具的特点。

CAI 软件切忌呆板或“黑板搬家”，否则，失去计算机辅助教学的意义。我们曾购过一批物理辅助教学软件，大多是单一的叙述物理定理和公式，同学们看后批评说：“只把计算机屏幕当作一块可以自动换内容的小黑板，形式单一，没有多大变化，除了可以加快速度外，别无它用……”

五、符合中小学生的心理特点

由于我们设计的 CAI 软件应用对象是中小学生，所以在形式上以丰富多采、灵活多变吸引学生。因为中小学生年幼、爱动，注意力不持久，受兴趣的支配性大。在设计软件时应符合青少年学生的心理特点。因此，CAI 软件应具备以下几个特点：

1）形式多样：如果始终重复同样的图形、动作及要求，就会使学生感到腻味。因此，要注意有强弱刺激的起伏，使得在一次强刺激下产生的兴趣和注意力开始转移之前，又及时给予一个新的强刺激，引导他进入另一个高潮。

2）形象生动：要利用计算机的绘图和音乐功能，设计出声图并茂，熔教学、娱乐为一体的软件，达到有声有色的动画效果。

3）连续性强，逐步深入：软件中所给出的教学内容及思维过程不能跳跃太大，并注意及时反馈、由浅入深，让学生自然地发现规律性。不要突然提出结果，也不要提出不切实际的要求，让学生感到束手无策。

4）可靠性：具有容错能力。

5）使用方便：教师调整灵活，学生控制灵活，屏幕提示简单明了。

6）程序技巧：应对算法进行优化，使程序有较高的效率。

7）商品化程度：应有较详细的功能说明、使用说明及必要的维护说明。

高分辨率图形的任意比例压缩

河北任丘华油三中高二　　何东强

在《学生计算机世界》总第八十三期上有一则求色码程序。经过实践，发现这则程序还可以用来压缩图形。程序的原理很简单，就是先将高分辨率第二页上对应点的颜色码求出，然后把该点的横纵坐标 X，Y 除以各自的比例数后，再画在高分辨率第一页上。

程序用 BASIC 语言编程，运行时，先要求输入压缩后的起始坐标（X_2，Y_2），接着输入横纵坐标的压缩比例（X，Y）。X，Y 的值越大，运行速度越快。如果能把 BASIC 程序编译成机器语言，运行速度会提高近九倍。

程序 1 使用机器语言程序 2，由于省去了计算，机器语言也相应变短，所以运行速度快。

程序 1

```
9 ONERR GOTO 100
10 INPUT"START(X2,Y2)";X2,Y2:IF X2<0 OR Y2<2
   THEN 10
20 INPUT"X,Y=";X,Y: IF X< 1 OR Y<1 THEN 20
30 HGR
40 X1=X:Y1=Y
50 FOR Y=0 TO 191 STEP Y1:FOR X=0 TO 279 STEP X1
60 POKE 230,64:& X,Y
70 POKE 230,32:HCOLOR= 3 * PEEK (0)
80 HPLOT X2 +X /X1,Y2+Y/ Y1
90 NEXT :PRINT Y:NEXT
100 END
```

程序 2

```
0300-  20  B9  F6  20  11  F4  A4  E5
0308-  B1  26  C9  80  B0  07  A2  00
0310-  86  01  4C  1B  03  A2  01  86
0318-  01  E9  80  25  30  F0  02  A9
0320-  01  85  00  60
```

中华学习机语音合成板的研制与应用

四川大学　计算中心　周凉　　计算机系　苟大举　贺德珏

摘要： 本文介绍了语音合成处理器 TMS5220 原理和由它组成的中华学习机语音合成板以及在中小学语音辅助教学中的应用。

一、引言

目前，语音合成技术已取得很大进展，已有若干种方法能够以很低的比特率合成高质量的语音，各种 LSI 语音合成蕊片(VSP)相继出现，语音合成技术在许多方面已经应用。本文介绍中华学习机利用 TMS5220 语音合成处理器构成的中华学习机语音合成输出板的原理，配上适当的词汇，介绍了它在语音教学方面的应用。

二、TMS5220 语音合成处理器简介

TMS5220 是美国德克萨斯仪器公司近年来推出的产品，图 1 是内部结构框图。

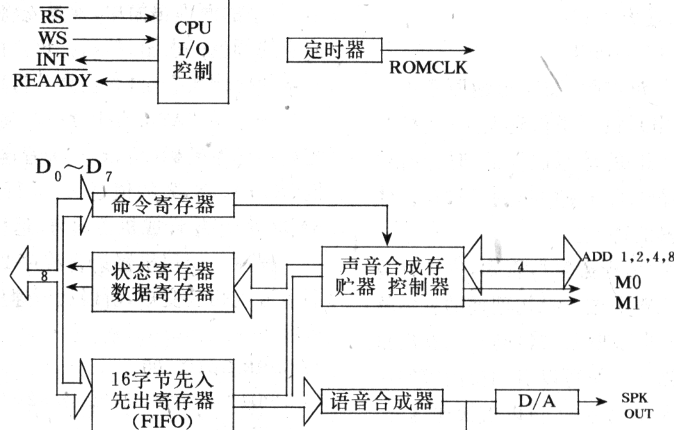


图1 TMS5220语音合成处理器内部结构框图

TMS5220 语音合成处理器的工作过程可简述如下：在复位或完成一段语音合成后，与之相连的计算机第一次送到接口线上的有效数据，语音合成处理器的内部控制逻辑自动地将此数据送往命令寄存器，然后再进行命令解码，若是外部说话命令，那么紧接着出现在接口线上的数据流便依次被送往先入先出寄存器(FIFO)，同时内部状态寄存器的内容也发生了相应的变化。合成单元在内部控制逻辑的控制下，按一定规律取出 FIFO 中的数据，并进行语音合成，合成结果得到语音的重建信号数据，这些数据经 D/A 转换器转换后得到供外部放大器用的模拟语音波形输出，以上合成过程一直延续到 FIFO 全空或者遇到仃止代码(即语音能量为二进制数 1111B)。本系统采用外部说话命令，接着送的数据必须是内存已知语音词汇表中待发音的语音数据。外部说话命令字为十六进制 $ 60。

TMS5220 有一个内部状态寄存器为 CPU 提供 VSP 的状态信息。状态寄存器有三个标志位：说话状态标志位 Ts，FIFO 半空状态标志位 BL(即 FIFO 内的数据是否少于或等于 8 字节，CPU 根据这一状态位，确定是否可以马上发送下一个字节语音数据)和全空状态位 BE，CPU 可根据需要访问这些标志位以决定程序的进程。

三、硬件原理

图 2 为中华学习机语音合成板硬件原理线路图。图中 TMS5220 的数据总线 D0～D7 与 6522 多用途接口适配器(VTA)的 PA 口连接(注意：TMS5220 的 D_0～D_7 与 6522 的 PA_0～PA_7 的连接是反序相连的，即两蕊片低位和高位交叉连接(见图 2)，这是语音编码数据所需要的)。读出线 $\overline{RS}$，写入线 $\overline{WS}$和准备就绪线 $\overline{READY}$分别与 6522 的 PB2、PB1、PB0 相连。由于驱动能力的需要，因此在 PA 口和 PB 口分别接上 10K 上拉电阻，TMS5220 的数模转换器输出与地之间接入 1.8K 电阻，是作为电流对电压的转换用的，输出信号经隔直电容加到 LF353 组成的低通有源滤波器输入端，最后经 LM386 音频放大器放大后直接推动小型扬声器。选取 6522 目的在于 TMS5220 为一慢速外部器件，因此 6522 作隔离用以配合时序要求，同时 6522 的定时器可作语音报时用。74LS08 用于 APPLE Ⅱ 与 6522 时钟相位配合。

四、软件设计

这里主要介绍讲一个单词的程序(见图 3)，考虑到 TMS5220 语音合成处理器是一种慢速的蕊片，并

且有比较严格的时序要求。中华学习机通过 6522 与 TMS5220 之间采用查询联络方式创造了必要条件。这里系统的查询主要包括两个方面：TMS5220 $\overline{READY}$ 线的电平和 TMS5220 的内部状态寄存器。每写一个语音数据均要查询这两个状态。TMS5220 的 $\overline{WS}$ 和 $\overline{RS}$ 由中华学习机通过 6522 的 PB1 和 PB2 用软件产生，以满足慢速要求。写一个语音数据（见图 4）和读 TMS5220 的状态（见图 5）分别为一个子程序。图 6 是 TMS5220 读写时序。

五、应用

有了上面发一个单词语音的子程序和语音合成板，再配上各种语音数据就可将中华学习机或 APPLE Ⅱ 机用于语音提示方面的应用，如 APPLE Ⅱ 语音报时系统、语音辅助教学系统等等。这里简单介绍一下用于语音辅助教学方面的应用：用中华学习机一边显示九九表，并同时读出声音。需要的语音词汇表包含数字 0～10 及“得”的单字语音数据，每个字的语音数据存于中华学习机内存中，同时建立一个数据起始地址的地址表，该地址以数字“0”的起始地址开头，按数字大小顺序排列，最后为单字“得”的起始地址。根据起始地址表就可找到要讲某个单词数据的起始地址，根据要发音的等式（如显示为 7×8＝56，则读为七八五十六，如果显示为 1×7＝7，则读为一七得七）反复调用发一个单字的子程序就可读出等式来。现已成功地编制了九九表显示发音程序，类似此思想可增加一些单词用于小学算术自动语音教学系统，即可用中华学习机训练小学生学数学，提高教学质量，很有实际意义。

注：本文语音数据取自四川大学模式识别室语音库。

参考文献

1. TMS5220 Voice Synthesis Processor Data Manual

2.《6502 微处理机及其应用》荣树熙、张开敬编，北京师范大学出版社，1984。

3.《苹果 Ⅱ BASIC 程序设计》张士英编，北京师范大学出版社，1985。

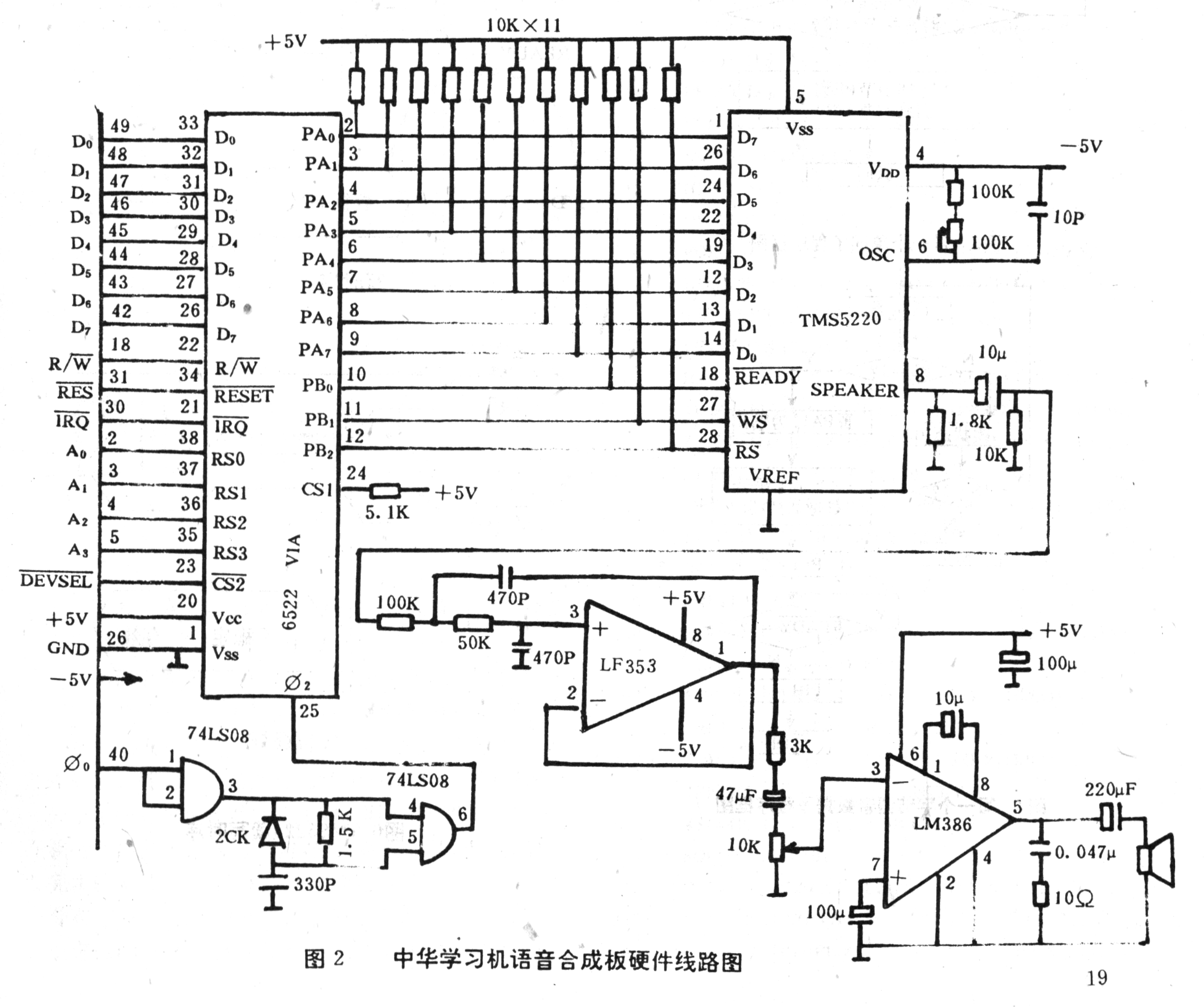


图 2　中华学习机语音合成板硬件线路图

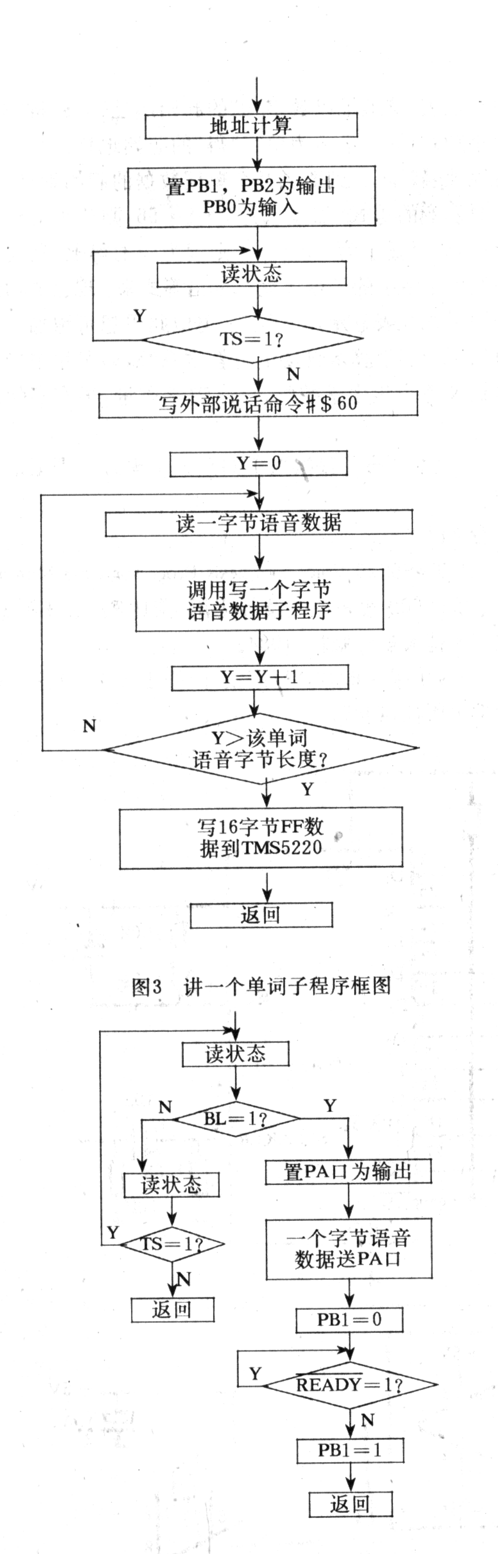


图3 讲一个单词子程序框图

图4 写一个字节语音数据子程序框图

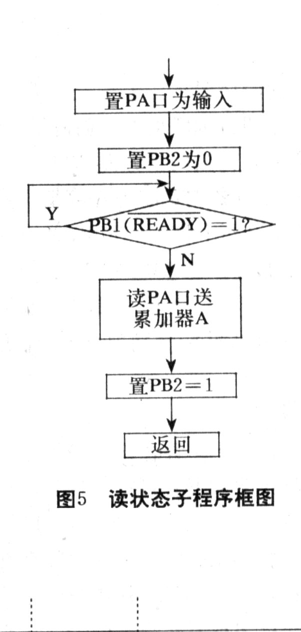


图5 读状态子程序框图

RS

WS

READY

$D_0 \sim D_7$

数据有效

写时序

WS

RS

READY

$D_0 \sim D_7$

不稳定

有效

读时序

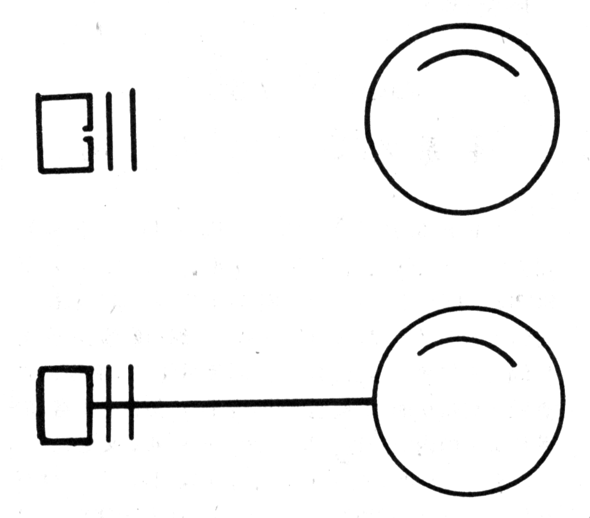
图6 TMS5220读写时序

麦克斯韦速率分布演示程序

内蒙古工学院物理教研室　王克勋

在普通物理教学中，分子运动的统计规律是一个教学难点。根据教学需要编制验证麦克斯韦速率分布的蔡特曼—葛正权实验[1]的模拟演示实验程序，用来辅助教学，效果良好。

为节省内存并使实验可连续进行，在磁盘上建立二个图形文件（GAS1、GAS2）和一个文本文件（GASTEXT）。GAS1 是实验装置(图一)，GAS2 是麦克斯韦速率分布曲线。作出这二个图形的程序很简单，不再列出。



图二

文本文件 GASTEXT 是为绘出检测装置动态旋转而存储的数据，由程序二建立。

本演示程序运行时，首先从磁盘调出文本文件，调出图形文件 GAS1 显示实验装置，按任一键，银原子在小炉中蒸发形成分子束从小孔射出(图二)，这时装置作顺时针转动，当装置的狭缝转到分子束入射方向时，分子束便穿过狭缝射到弯曲玻璃屏上。装置继续旋转，狭缝偏离分子束入射方向，分子束被遮断，至此，模拟完实验全过程。由子程序作出屏上分子按速率大小的分布，该分布由麦克斯韦速率分布函数 $f(v)=Av^2e^{-Bv^2}$ 决定。点由右向左随机作出，表现分子打在屏上的动态过程。点密度便表现出分子束的“速度谱”。然后，调出图形文件 GAS2，显示出该分布的对应曲线。这样，便由定性到定量，完成对麦克斯韦速率分布的数学演示。分别敲入“1”，“2”两键，可轮换显示动态过程图和分布曲线图，便于两相对照，敲入“0”，结束演示。

附：**程序一**

```
100  DIM X(8,101),Y(8,101)
110  D$= CHR$ (4)
132  VTAB 10:HTAB 15:PRINT "Please wait……"
138  PRINT D$;"OPEN GASTEXT"
140  PRINT D$;"READ GASTEXT"
142  FOR I=1 TO 80: FOR J=1 TO 101
144  INPUT X(I,J),Y(I,J)
146  NEXT:NEXT
150  HGR2:HCOLOR=3
160  FOR A=1 TO 279:HPLOT A,1 TO A,159:NEXT
170  HCOLOR=0:FOR A=2 TO 278:HPLOT A,2 TO A,
     158:NEXT : HCOLOR=3
180  PRINT D$;"BRUN GAS1"
200  HPLOT 20,70 TO 22,70 TO 22,55 TO 0,55 TO 0,89
     TO 22,89 TO 22,74 TO 20,74:HPLOT 30,55 TO 30,
     71:HPLOT 30,73 TO 30,89:HPLOT 40,55 TO 40,
     71:HPLOT 40,73 TO 40,89
220  DEF FN A(I)=180+R* COS(3.14*I/180):DEF
     FN B(I)=72+R*SIN(3.14*I/180)
225  INPUT A$
226  PRINT
230  FOR I=1 TO 200:HPLOT INT (RND(I)*22),INT
     (RND(I)*34+55):NEXT
240  FOR X=20 TO 30:FOR I=1 TO 4:HPLOT X,INT
     (RND(I)*9+69):NEXT:NEXT
250  FOR X=30 TO 40:FOR I=1 TO 2:HPLOT X,INT
     (RND(I)*4+71):NEXT:NEXT
260  XS=40:XT=140:GOSUB 660
320  FOR I=1 TO 101
330  IF I=50 THEN XS=140:XT=208:GOSUB 660
340  IF I=60 THEN XS=140:XT=208:GOSUB 720
350  HCOLOR=0:HPLOT X(1,I),Y(1,I) TO X(2,I),Y
```

```
     (2,I):HPLOT X(7,I),Y(7,I) TO X(8,I),Y(8,I)
360  HCOLOR=3:HPLOT X(3,I),Y(3,I) TO X(4,I),Y
     (4,I):HPLOT X(5,I),Y(5,I)
370  Z=PEEK(-16336)
380  NEXT
390  GOSUB 790
395  GET A$:CLEAR:PRINT
400  D$=CHR$(4):PRINT D$;"BRUN GAS2"
405  POKE -16301,0
407  VTAB 22
410  INPUT"HGR OR HGR2 OR END(1/2/0)";Q$
420  IF Q$="0" THEN TEXT:END
430  IF Q$="2"THEN POKE-16302,0:POKE-16299,0:
     GOTO 410
440  IF Q$="1"THEN POKE-16301,0:POKE-16300,0:
     GOTO 410
450  GOTO 410
660  HCOLOR=3
670  FOR X=XS TO XT
680  FOR K=1 TO 2
690  Y=70+4*RND(K)
700  HPLOT X,Y:NEXT:NEXT
710  RETURN
720  HCOLOR=3:R=40
730  FOR K=180 TO 190
740  HPLOT FN A(K),FN B(K):NEXT
750  HCOLOR=0
760  FOR K=140 TO 207
770  PHLOT K,68 TO K,75:NEXT
780  RETURN
790  HGR2:HCOLOR=3
800  HPLOT 1,20 TO 1,140 TO 260,140 TO 260,20 TO 1,
     20
810  FOR X=260 TO 1 STEP -1
820  X1=(X/80)^2/EXP((X/80)^2)*100
830  FOR I=1 TO X1
840  Y=20+120*RND(I)
850  Z=PEEK(-16336)
860  HPLOT X,Y:NEXT:NEXT
870  RETURN
```

程序二

```
10   DIM X(8,101),Y(8,101)
15   D$=CHR$ (4)
20   DEF FN A(I)=180+R*COS(3.14*I/180):DEF
     FNB(I)=72+R*SIN(3.14*I/180)
30   FOR I=1 TO 101
40   R=50
50   X(1,I)= FN A(130+I):Y(1,I)=FN B(130+I)
60   X(2,I)=FN A(131+I):Y(1,I)=FN B(131+I)
70   X(3,I)=FN A(120+I):Y(3,I)=FN B(120+I)
80   X(4,I)=FN A(121+I):Y(4,I)=FN B(121+I)
85   R=30
90   X(5,I)=FN A(310+I):Y(5,I)=FN B(310+I)
100  X(6,I)=FN A(311+I):Y(6,I)=FN B(311+I)
110  X(7,I)=FN A(250+I):Y(7,I)=FN B(250+I)
120  X(8,I)=FN A(251+I):Y(8,I)=FN B(251+I)
130  NEXT
140  PRINT D$;"OPEN GASTEXT"
150  PRINT D$;"WRITE GASTEXT"
160  FOR I=1 TO 8:FOR J=1 TO 101
170  PRINT X(I,J);",";Y(I,J):NEXT :NEXT
180  PRINT D$;"CLOSE GASTEXT"
```

参考文献

(1)程守洙　江之永编　普通物理学(1982修订本)　高等教育出版社　　P259

·实用程序·

二维实型数组的高速排序

西安高级中学　陆裕元

排序速度的快慢，有时直接影响着一个程序的质量，因此，如何加快排序速度是提高程序质量的一个关键。目前已有不少用机器语言实现数组高速排序的例子，本刊89年第6期刊载的《APPLE汇编语言高速排序》就是一例，但遗憾的是这些程序都只能用于一维实型数组的高速排序，对二维数组却无能为力，而很多的统计工作中却常常需要对二维实型数组排序(如学生成绩排名次)，针对这一缺陷笔者深入分析了多维数组在机内的存贮方式，编制了如下机器语言程序，排序时，只需用一条命令 & 实型数组名(a,0)就可实现对二维数组的高速排序(a=1或0)，本程序的特点是排序速度快，调用方便，经验证，对A(60,7)数组排序仅需3秒。

使用本程序对二维实型数组按某一关键元素高速排序时，对关键元素的存放位置有一定要求，现以学生成绩排名次为例(参看所附的BASIC程序)说明如下：假定有N名学生，考M−1门课，则可定义一个二维数组A(N,M)，其中A(N,1)存考生学号，A(N,2)，A(N,3)……A(N,M)存考生各门单科成绩，如果按总分排名次，则必须将总分存在A(N,0)，这是排序的关键元素。这样，按总分排序时，只需用&A(1,0)即可。如果按某一单科成绩排序，则必须将该单科成绩存在A(N,0)，排序仍用&A(1,0)。所以，用户使用时一定要注意将待排序的关键元素存在A(N,0)这样的位置上。

本程序是按降序排序，若按升序排序，则需修改$9580单元的值为$01，同时将BASIC程序中的调用命令改为 & 实型数组名(0,0)。另外在调用前，必须给指定单位元予置值。本文所附BASIC程序说明了正确的调用方法。BASIC程序各句意义说明如下：

第7句：设定 & 语句转向执行的地址。

第15句：将数组第一维上限值+1存入指定的两个单元$1A和$1B。

第17句：将数组第二维上限存入指定单元$95AB，并将第一维上限值+1乘5的值存入指定的四个单元$95B8，$95B2，$95C5，$95BF。

第20～40句：产生并输出原始的二维数组元素。

第50句：按总分排序。

第60句：输出排序后的二维数组元素。

其中7,15,17,50这几句是用户排序时必须加入的语句，不能缺少。

所附执行结果显示了9个学生考5门功课按总分排序的情况，其中第1列为学号，第2列至第6列为5门功课的单科成绩，第7列为总分。

本程序稍作改动还可实现三维、四维甚至多维数组按某一关键元素高速排序。

```
9500-  20  E3  DF  85  FA  85  FC  84
9508-  FB  84  FD  A9  01  8D  94  95
9510-  A9  00  8D  95  95  F0  11  18
9518-  A5  FA  69  05  85  FA  85  FC
9520-  A5  FB  69  00  85  FB  85  FD
9528-  A5  FA  A4  FB  20  F9  EA  AE
9530-  94  95  AD  95  95  8D  96  95
9538-  20  69  95  E8  D0  03  EE  96
9540-  95  E4  1A  D0  F3  AD  96  95
9548-  C5  1B  D0  EC  18  AD  94  95
9550-  69  01  8D  94  95  D0  03  EE
9558-  95  95  AD  94  95  C5  1A  D0
9560-  B6  AD  95  95  C5  1B  D0  AF
9568-  60  18  A5  FC  69  05  85  FC
9570-  90  02  E6  FD  A5  FC  A4  FD
9578-  86  1E  20  B2  EB  A6  1E  C9
9580-  FF  F0  14  60  A0  04  B1  06
588-   48  B1  08  91  06  68  91  08
9590-  88  10  F3  60  0A  00  00  A5
9598-  FA  85  06  A5  FB  85  07  A5
95A0-  FC  85  08  A5  FD  85  09  20
95A8-  84  95  A9  06  85  1F  18  A5
95B0-  06  69  32  85  06  A5  07  69
95B8-  00  85  07  18  A5  08  69  32
95C0-  85  08  A5  09  69  00  85  09
95C8-  20  84  95  C6  1F  D0  DF  A5
95D0-  FA  A4  FB  4C  F9  EA  FF
```

```
6    HIMEM:38143
7    POKE 1014,0:POKE 1015,149
```

```
10      INPUT "N,M=";N,M:DIM A(N,M)
15      POKE 27,INT((N+1)/256):POKE 26,N+1-
        PEEK (27) * 256
17      POKE 38315,M:POKE 38328,INT ((N+1) *
        5/256): POKE 38322, (N + 1) * 5- PEEK
        (38328) * 256:POKE 38341,PEEK(38328):
        POKE 38335,PEEK(38322)
20      FOR I=1 TO N:A(I,0)=0:A(I,1)=I:PRINT
        I;"";:FOR J=2 TO M
30      A(I,J)=(INT(RND(1) * 60)+40):PRINT A
        (I,J);"      ":A(I,0)=A(I,0)+A(I,J)
40      NEXT:PRINT A(I,0):NEXT:PRINT
50      CALL-198:&A(1,0):CALL-198
60      FOR I=1 TO N:PRINT A(I,1);"   ";:FOR J=
        2 TO M:PRINT A(I,J);"      ";:NEXT:PRINT
        A(I,O):NEXT:END
RUN
N,M=9,6
1       62 52 91 88 64 357
2       96 77 95 60 90 418
3       96 58 45 89 43 331
4       93 97 93 98 67 448
5       50 85 56 62 99 352
6       55 59 80 83 45 322
7       52 87 89 94 51 373
8       91 63 75 97 44 370
9       94 96 78 51 59 378

4       93 97 93 98 67 448
2       96 77 95 60 90 418
9       94 96 78 51 59 378
7       52 87 89 94 51 373
8       91 63 75 97 44 370
1       62 52 91 88 64 357
5       50 85 56 62 99 352
3       96 58 45 89 43 331
6       55 59 80 83 45 322
```

·实用程序·

LOGO的磁带存贮方法

武汉 郑华强

MIT LOGO 本来是磁盘高级语言，CEC—I 机已固化在 ROM 中，以磁盘为外存。能否以磁带为外存呢？笔者通过分析，了解到 LOGO 的编辑区首址为 $2000，尾址则由零页单元 $9C 和 $9D 构成的编辑区指针指示。因此，只要将编辑区内容和编辑区指针用磁带保存起来，即达到了以磁带为外存的目的。据此，设计了两套磁带存贮方法：第一套是原始方法，只涉及到监控级的命令（过程见附录一）。第二套是通过四个 LOGO 过程（见附录二）自动实现磁带存贮（见附录三）。经过试用，效果很好。

附录一：LOGO 磁带存贮（一）

①存入磁带

步骤 1°？EDIT↙　将工作空间装入编辑区这里以附录二所示编辑区文本为例。

步骤 2°

打入 ctrl—G 直接退出编辑器（可以对文本做适当修改或删节后再退出）。

步骤 3°

？. BPT↙　进入监控状态

＊9C. 9D↙　查询编辑区指针

如显示：009C—80　21 则表示编辑区尾址是 $2180（这里以附录二的文本为例）

准备好录音，然后打入＊2000. 2180W↙将编辑区内容存入磁带，并在带盒上记下 $2180

步骤 4°

＊ctrl—y↙　返回 LOGO 状态，存带结束。

（磁带上这四个过程的文本名为. S. L. RCD 文件）

②从磁带上取出过程

步骤 1°

？. BPT↙　进入监控状态

步骤 2°

带盒上记着编辑区尾址 $2180，因此打入＊9C：80　21↙　恢复编辑区指针

准备好放音

＊2000. 2180R↙　恢复编辑区内容＊ctrl-y↙，返回 LOGO 状态

步骤 3°

？EDIT↙　将显示如附录二所示文本

打入 ctrl—C 处理编辑器，使从磁带装入的文本进入工作空间。取带结束。

附录二：

```
TO . SAVE
SAVE. RCD 156 0 157 0(存指针)
SAVE.  RCD  0  32  . EXAMINE  156  .
EXAMINE 157(存编辑区)
END
TO SAVE. RCD : BL : BH : EL : EH
. DEPOSIT 60 : BL. DEPOSIT 61 : BH
. DEPOSIT 62 : EL. DEPOSIT 63 : EH
. CALL 53533 60 (53533 为监控存带程序入口)
END
TO. LOAD
LOAD. RCD 156 0 157 0(取指针)
LOAD.  RCD  0  32.  EXAMINE  156.
EXAMINE 157(恢复编辑区)
END
TO LOAD. RCD : BL : BH : EL : EH
. DEPOSIT 60 : BL . DEPOSIT 61 : BH
. DEPOSIT 62 : EL . DEPOSIT 63 : EH
. CALL 53590 60(53590 为监控取带程序入口)
END
```

附录三：LOGO 磁带存贮（二）

步骤 1—1°　（先执行？ER　ALL↙）

？TO↙　进入编辑器

按附录二中的文本输入，然后打入 ctrl-C 处理编辑器，工作空间中定义出四个新过程。

步骤 1—2°　（此步骤只需在第一次存带或复制此四个过程时执行）

按附录一的方法存磁带备用。

步骤 2—3°　（刚开机或以上四个过程需恢复时执行）按附录一的方法从磁带中恢复这四个过程。

步骤 3—1°　（当以上四个过程在工作空间中时）

①存带方法：

？EDIT↙　将工作空间装入编辑器，

然后删掉不必要的内容，打 ctrl—G 退出。

准备好录音

？. SAVE↙

光标消失，十几秒后产生"嘟"的一声，存入编辑区指针，过一会儿又响第二声。且光标重现，存带结束。

· 实用程序 ·

低分辨率全屏幕绘图

黄文贵

已有不少文章介绍过中华学习机高分辨率全屏幕绘图的方法。笔者认为用键盘在低分辨率下进行全屏幕绘图也是行之有效的。

程序运行后，首先进入中文说明。按任意键进入画图状态，画图结束后按《ESC》键又转入中文状态，根据中文提示，用户自行选择重画、存盘、打印、结束。

存盘和打印的程序省略，请读者参阅 88 年第 6 期《苹果园》《APPLE 机低分辨图形的打印方法》，进行存盘和打印。我是把低分辨图形转入到高分辨率后，再进行存盘和打印的。

```
10  PRINT  CHR$(4);"PR#3":PRINT:HGR2
20  VTAB2:HTAB 8:PRINT"低分辨率全屏幕绘图":PRINT
30  PRINT  TAB(7);"I-向上";:HTAB 20:PRINT"Y-左上"
40  PRINT  TAB(7);"M-向下";:HTAB 20:PRINT"G-左
    下"
50  PRINT  TAB(7);"J-向左";:HTAB 20:PRINT"H-右上"
60  PRINT  TAB(7);"K-向右";:HTAB 20:PRINT"B-右下"
70  PRINT TAB(7);"7-画图";:HTAB 20:PRINT"0-擦图"
80  VTAB 10:HTAB 7:PRINT"Esc 键结束画图";:GET K$
90  X=0 : Y=0 : XI=39 : YI=47
100  TEXT : GR
110  POKE-16304,0 : POKE-16302,0 : POKE-16300,0 :
     POKE-16298,0
128  FOR I=40 TO 47
130  FOR J=0 TO 39
140  COLOR=0 : PLOT J,I
150  NEXT J,I
160  GET A$ : A=ASC(A$):E$=A$
170  IF A=ASC("0")OR A=ASC("7")THEN E=VAL(E
     $):GOTO 160
180  IF A=73 AND Y>0 THEN Y=Y-1 : GOTO 260
190  IF A=74 AND X>0 THEN X=X-1 : GOTO 260
200  IF A=75 AND X<X1 THEN X=X+1 : GOTO 260
210  IF A=77 AND Y<Y1 THEN Y=Y+1 : GOTO 260
220  IF A=71 AND X>0 AND Y<Y1 THEN X=X-1 : Y=
     Y+1 : GOTO 260
230  IF A=72 AND X<X1 AND Y>0 THEN X=X+1 : Y
     =Y-1 : GOTO 260
240  IF A=89 AND X>0 AND Y>0 THEN X=X-1 : Y=Y-
     1 : GOTO 260
250  IF A=66 AND X<X1 AND Y<Y1 THEN X=X+1 : Y
     +1 : GOTO 260
260  COLOR=15 : PLOT X,Y : FOR I=1 TO 30 : NEXT I :
     COLOR=E : PLOT X,Y
270  IF A<>27 THEN 160
280  TEXT : PRINT CHR$(4);"PR#3":PRINT : HGR2
290  VTAB 3:HTAB 7:PRINT"1.重画";:HTAB 20:PRINT"
     3.打印"
300  VTAB 5:HTAB 7:PRINT"2.存盘";:HTAB 20:PRINT"
     4.结束"
310  VTAB 8:HTAB 13 : PRINT "请选择:";:GET K$:
     PRINT K$:K=VAL(K$)
320  IF K<1 OR K>4 THEN 310
330  IF K=1 THEN 10
340  IF K=2 THEN PRINT"用户自行设计"
350  IF K=3 THEN PRINT"用户自行设计"
360  IF K=4 THEN END
```

---

②取磁带内容方法：准备好放音

附录三

?.LOAD↙

光标消失，十几秒后产生“嘟”的一声，恢复编辑区指针，过一会儿又响第二声，且光标重现，取带结束。

这时编辑区已经恢复。

? EDIT↙　进入编辑器

可以看到从磁带取出的文本，然后打 ctrl—C 处理，新的过程即被定义装入工作空间。

附录四：　注意事项

一、刚开机时，立刻用附录一中的方法取出 S. L. RCD 文件后，进入编辑器，不一定会看到附录二所示文本，可能是一片空白或一些奇怪的东西。这时打 ctrl—C，附录二的四个过程亦会定义出来。若这时再进编辑器，附录二的文本即会出现。

附录三中 LOAD 命令使用时也有可能出现以上情况。

二、在附录三的步骤 2—3°时，最好在监控状态打入

*3F2:F8　03　A6 ↓

使 Reset 软矢量指向监控命令 ctrl—Y ↓ 的入口，于是 ctrl—Reset 的作用成为返回 LOGO。

三、LOAD 命令在执行中时，若出现取带错误，将造成机器锁死，这时可用 ctrl—Reset 返回 LOGO 状态。

四、只有 S. L. RCD 文件需用附录一的方法处理，其它文件皆可用附录三的方法处理。此方法可为没有驱动器的用户提供极大方便。

·实用程序·

键盘录入综合训练与考核程序

柳州市一职高　　张耀洲

键盘录入速度是计算机操作员录入员技术水平的一个重要指标。如何进行录入技术的训练，直接影响操作员录入员的培训速度。目前见诸报刊、书籍的训练程序和考核方法都有诸多不便。例如参考文献①所用的综合训练与考核方法，就具有代表性。其主要缺点是：要用马表计时，不能自动计时、计算成绩；要逐个逐句校核。因此，其中规定的考核标准很难判定是否真正达到了。考核时，既麻烦费时，考核成绩也难免会有一定的水分。

下面所列几个程序克服了上述缺点，自动计时，自动计算成绩，按错键可以自动倒扣若干个。

程序 1 为 CEC—I 上的程序。运行后，将键盘上的五十九个字符随机出现排列成 11 行 40 列（隔一行显一行）。按任一键后屏幕上方显示电子表。立即开始按键。一行中正确按键十次后，按错一键时，光标左移 D（=10）个字符，重新按顺序按键。本程序允许最大输入速度为 503 个/分钟，若删去 70 行的 FLASH 及 80 行的 NORMAL，并且删去显示电子表的语句，则允许的最高输入速度还可提高。但是，本程序 90 行以前的任何修改都有可能影响电子表的精确度。修改了就必须重新测定 MI 和 SI（=MI/60）的值。考核与练习时间为 E（=5）分钟，满 5 分钟后，自动显示成绩，E 最大允许 $\frac{440\times40}{V}$分钟，V 为录入速度，单位是个/分钟。例如 V=250，则 E≤68.4 分钟。即允许输入速度为 250 个/分钟的人连续练习 68.4 分钟。若这个指标还要加大，将 80 行最后一句中的 L<F 的 F 改为大于 40 的数即可。

训练或考核过程中，允许连续五次（每次 5 分钟）输入，满五次时，自动显示各次成绩，如限定每人只考三次，则可将 92 行中“CU>4”改为“CU>2”即可。这样，考核时，只要规定学生在指定时间（例如半小时）内拿出成绩来，然后老师再去看即可，无须守在学生旁边。

倒扣个数 D、考核时间 E 的值可在 10 行中修改。

程序 2 为 LASER—310 机的程序。运行后屏幕上也是显示满屏（15 行×32 列）480 个随机排列的字符。按任一键后就应开始输入，电子表不显示出来。允许最高输入速度为每分钟 272 个字符。达到最高速度时电子表每 5 分钟慢 30 秒；若 5 分钟内不按任何键，则电子表快 20 秒。这种计时误差引起的成绩误差已通过 92 行的 K 值计算公式得到纠正。因此，显示出来的成绩，若不是每分钟 132 个时，5 分钟的总数与每分钟平均数不一致。而且，92 行以前的任何改动都可能造成考核成绩的不准确。程序 2 运行后输入超过 F(=5)个字符后，按错一键，倒扣 F(=5)个。如要修改这个值可改 30 行 F 的值。

程序 3 为 COMX—PC1 机程序。运行后按空格键开始计时录入，应按闪烁的字符键。一行中正确按键超过 S(=21)次后，按错一键，倒扣 E(=10)个。这两个值都可以在 20 行修改。但必须使 B<S. 每练习一次为 M(=5)分钟，M 的值可在 20 行修改。除了 80 行与 M 值外，200 行以前的任何改动都不会影响电子表的准确性，而只会影响允许最高输入速度。本程序允许最高输入速度为 255 个/分钟。运行后允许连续操作

$\frac{(R+1)*E*40}{V}$分钟

其中 E(=22)为一屏的行数，V 为输入速度，例如 V=250，按程序中的 R=1.0，E=22 计算，则运行程序 3 可以连续输入大约 39 分钟。若要加大这种值，可以在 10 行修改 R 的值. 程序 3 也可以连续考核五次，满五次后，同时显示各次的成绩。按 ESC 中断与键入 RUN 命令后重新运行也不会冲掉前几次的成绩。

以上三个程序都是将 59 个键盘字符以同等机率随机显示在屏幕上的，因此，很适合初期综合训练。事实证明，我校 88 微机中专班 35 个学生仅经过 16 周，每周 1、5——2 小时的练习（其中还包括单个字符键的输入练习 7—9 周）即全部达到每分钟 80 个以上的水平，16 人超过 90 个，9 人超过 98 个，最高的达到 110 个。全班学生仅用 6 台机器一个下午的时间就基本考核完。

程序 1

```
10 Z=59:J1=32:C=2:TE=18:N=10:F=40:JJ=49152:
   QJ=49168:D=10:OT=127:OS=OT+1:E=5:TF=
   34:S1=8.38:M1=503
20 Q=(N+1)*F:DIN A%(Q)
25 L=0:B=C:S=0:M=0:HOME:FOR I=1 TO Q:A%(I)=
```

```
RND(1) * Z+J1:VTAB B:PRINTCHR $ (A%(I));:B=B
+(I=INT (I/F) * F) * C:NEXT:GET W $ :PRINT"";
30 B=1:FOR I=0 TO N:B=B+C:FOR K=1 TO F:J=I * F
+K:VTAB B:HTAB K:FLASH:PRINT"^ ";:P= PEEK
(JJ):R=(P>OT):POKE QJ,0:P=P-OS:T=((P=A%
(J))-(P<>A%(J)AND K>D) * D) * (R=1):K=K-1
+T:IF M=E THEN 90
80 NORMAL:VTAB 1:HTAB TE:PRINT M;"'";INT(S/S1);
CHR $ (TF);:S=(S+1) * (S<M1):M=(S=0)+M:
VTAB B:HTAB(T>-1) * K+(T<0) * (K-T+1):PRINT"
";:NEXT:NEXT:L=L+1:IF L<F THEN 30
90 NORMAL:INVERSE:VTAB 2:HTAB 8:PRINT M;"'=";L
* (N+1) * F+J-1;TAB(23);"ZHENGJI:";:Z1=INT
((L * (N+1) * F+J-1)/M):PRINT Z1:POKE QJ,0:CU
=CU+1:Z(CU)=Z1
92 IF CU>4 THEN 250
95 FOR I=1 TO 30:PRINT CHR $ (7);:NEXT:PRINT"
SHIFOU JISU? (Y/N)":NORMAL
96 GET W $ :IF W $ ="Y"THEN 25
100 IF W $ ="N"THEN 250
110 GOTO 96
250 FOR I=1 TO CU:VTAB 7+I:HTAB 7:PRINT "DI";I;"
CU:"; Z (I): NEXT: PRINT" D =";D: PRINT CHR
$ (7);:GOTO 250
```

程序 2

```
5  REM LASER-310 JI ZHIFA ZHENGXI WU DANZU BIAO
DAO KO F GO
10 CLS:N=480:P=28704:DIM A(N+2):FOR I=1 TO N:A
(I)=RND(59)+31:K=0
20 POKE P,A(I):P=P+1:NEXT:A(0)=32:L=0:S=0
30 F=5:P=28703:B=60:C=4:T=34:E=5:G=159:H=
11:R=9
33 IF INKEY $ < " " OR INKEY>"X"OR INKEY $ =""
THEN 33
35 FOR I=1 TO N:K=(K+1) * ABS(K<C):S=(S+ABS(K
=C)) * ABS(S<B)
40 A $ =INKEY $ :U= ABS(A $ <>""):A $ =A $ +CHR
$ (R):Q=-(U * ASC(A $ )=A(I+1))
45 J=-(U=1) * J+1:Q=Q+F * (U=1 AND Q=0 AND J
=2 AND I>F)
50 M=M-(S=B):POKE P+I,G * ABS(Q>-1)-A(I) * (Q<
0):IF M=E THEN 90
60 D =N+I-1:D=D-INT(D/N) * N:POKE P+I-1,A(D):I
=I+Q+(I<>N)
80  NEXT:L=L+1:IF L<10 THEN POKE P+N-1,A(N-
1):POKE P+N,A(N):GOTO 35
90  CLS:V=(I+L * N)/M:PRINT @45,M" FENZHONG
="I+L * N
92  K= 300/280+(1/330-1/280) * V:PRINT @77,"
ZHENGJI:"; INT (K * V)
93  PRINT @99,"F="F":PRINT @131,"LZF-F-9025"
95  FOR J=1 TO 10:SOUND23,7:NEXT
100 PRINT @163,"SHIFU JISU? (Y/N)":IF INKEY $ ="Y"
THEN RUN
110 IF INKEY $ ="N" THEN END
120 GOTO 100
```

程序 3

```
5 REM CZF>20-10 900208
10 TIMOUT200:C POS (0,0):CLS:DEFINT A:E=22:F=40:
R=10:W=64:G=F+1:DIM A(E,F),Z(10)
20 P =59:Q=32:B=10:S=21:M=5:C $ =CHR $ (107):
FOR I=1 TO E:FOR J=1 TO F:A(I,J)=RND (P)+Q
30 IF A(I,J)=W THEN J=J-1:GOTO 50
40 PRINT CHR $ (A(I,J));
50 NEXT:NEXT:L=0
65 IF KEY=32 THEN 80
70 GOTO 65
80 TIME(M * 3000)
83 FOR K=0 TO R:FOR I=1 TO E:FOR J=1 TO F:A=
KEY:C POS(I-1,J-1):PRINT CHR $ (A(I,J)):IF A=0
GOTO 120
90 IF A=A(I,J) THEN J=J+1:GOTO 120
100 IF J<S GOTO 125
110 J=J-B
120 IF J=G GOTO 130
125 C POS(I-1,J-1):PRINT C $
130 J=J-1:NEXT:NEXT:NEXT:K=K-1:I=I-1:J=J-1
200 C POS(0,0):CLS:X=(K * E+I-1) * F+J:PRINT M;"
FENZHONG=";X;" ZHENGJI:"; INT(X/M)
201 C POS (2,7):PRINT"S=";S:C POS(3,7):PRINT "B
=";B:Y=Y+ 1:Z(Y)=INT (X/M)
210 MUSIC(5,4,15):WAIT (150):MUSIC(0,0,0)
216 IF Y<4 GOTO 260
220 PRINT "SHIFOU JISU? (Y/N)"
230 IF KEY=89 THEN RUN
250 GOTO 230
260 FOR L=1 TO Y:C POS (L+7,7):PRINT "DI";L;;"
CU:";Z(L):NEXT
265 C POS (7+L,7):PRINT "S=";S:C POS(8+L,7):
PRINT"B=";B
270 IF KEY=1 THEN Y=0:RUN
280 GOTO 270
300 REM COMX—PC1 ZHIFA 900208
```

参考文献

①刘尊全　主编　微型计算机操作员录入员培训教材　清华大学出版社



打印机命令的再定义

山东曲阜师范大学物理系　狄振强

通常，利用打印机打印图形或程序清单时，一般要进行多步操作，诸如接通打印机、设置打印方式、启动打印、断开打印机等等。这样的操作不仅麻烦，而且还将一些不必要的字符也打印下来，如打印时设置的一些命令象 POKE 1913，n、PRINT CHR $ (17)、POKE 1657，n、LIST、PR＃0 等等。这些多余的字符使得整个纸面既不整洁也不美观。

为了减少操作上的麻烦，避免不必要的字符的出现，笔者用 6502 指令编写了一段程序，定义了新的打印命令。利用它可完成计算机的自动打印，使得打印的控制操作变得非常简便。

打印程序通过 & 字符特有的内存转向功能，使程序进入执行。

下面介绍打印控制命令

1. 图形打印命令

若进行图形打印，可通过下面的操作命令来完成。

①&P1　缩小打印高分辨率图形第一页。

②&P2　缩小打印高分辨率图形第二页。

③&P3 联页打印高分辨率图形第一页及第二页。

④&P33　缩小反相打印高分辨率图形第一页。

⑤&P34　缩小反相打印高分辨率图形第二页。

⑥&P65　放大打印高分辨率图形第一页。

⑦&P66　放大打印高分辨率图形第二页。

⑧&P97　放大反相打印高分辨率图形第一页。

⑨&P98　放大反相打印高分辨率图形第二页。

2. 程序清单的打印命令

①&L　打印全部程序清单。

②&Ln　打印行号为 n 的语句。

③&Ln$_1$，n$_2$ 打印行号为 n$_1$ 至 n$_2$ 间的程序段。

④&L，n 打印行号为 n 以前的程序段。

⑤&Ln，打印行号为 n 以后的程序段。

⑥&L：m 设置打印宽度为 m，打印全部程序清单。

⑦&Ln：m 设置打印宽度为 m，打印行号为 n 的语句。

⑧&Ln$_1$，n$_2$：m 设置打印宽度为 m，打印行号为 n$_1$ 至 n$_2$ 间的程序段。

⑧&L，n：m 设置打印宽度为 m，打印行号为 n 以前的程序段。

⑩&Ln，：m 设置打印宽度为 m，打印行号为 n 以后的程序段。

这里，& 以后的字符 p 代表图形打印，L 代表程序清单打印，冒号后面的数值是设置的打印宽度。

打印程序可以在开机之后或打印之前任何时候调入内存，操作为：BRUN 打印程序（注意不能用 BLOAD）。

欲进行打印时，只要输入上面的命令，计算机便自动开启打印机并进行打印，打印结束后将自动关闭打印机，以后的操作只在屏幕上输出。这些新定义的打印命令如同计算机的外部命令一样，在操作中犹为简便、实用。

打印机自动打印程序（机器码）

6000-	A9	0B	8D	F6	03	A9	60	8D
6008-	F7	03	60	A0	00	A9	00	85
6010-	3C	A9	02	85	3D	A9	40	85
6018-	3E	A9	02	85	3F	A9	50	85
6020-	42	A9	02	85	43	20	2C	FE
6028-	AD	51	02	C9	50	F0	05	C9
6030-	4C	F0	38	60	A0	00	A9	CB
6038-	85	3C	A9	60	85	3D	A9	EF
6040-	85	3E	A9	60	85	3F	A9	00
6048-	85	42	A9	02	85	43	20	2C
6050-	FE	A0	00	B9	52	02	C9	20
6058-	D0	03	4C	67	60	C9	00	D0
6060-	03	4C	B6	60	99	0E	02	C8
6068-	4C	53	60	A0	00	A9	F1	85
6070-	3C	A9	60	85	3D	A9	17	85
6078-	3E	A9	61	85	3F	A9	00	85
6080-	42	A9	02	85	43	20	2C	FE
6088-	A0	00	B9	52	02	C9	00	D0
6090-	03	4C	B6	60	C9	3A	D0	03
6098-	4C	A2	60	99	16	02	C8	4C
60A0-	8A	60	A2	00	B9	53	02	C9
60A8-	00	D0	03	4C	B6	60	9D	0E
60B0-	02	C8	E8	4C	A4	60	A2	01
60B8-	86	B9	A2	FF	86	B8	20	B1
60C0-	00	20	59	D5	20	B1	00	4C
60C8-	28	D8	00	50	52	23	31	3A
60D0-	50	4F	4B	45	31	39	31	33
60D8-	2C	20	20	20	3A	50	52	49
60E0-	4E	54	43	48	52	24	28	31
60E8-	37	29	3A	50	52	23	30	00
60F0-	00	50	52	23	31	3A	50	4F
60F8-	4B	45	31	36	35	37	2C	34
6100-	30	20	3A	4C	49	53	54	20
6108-	20	20	20	20	20	20	20	20
6110-	20	20	3A	50	52	23	30	00

·趣味程序·

"苹果"摇奖机

上海市南汇县教师进修学校　　刘同侃

利用计算机的随机函数，可以方便地进行开奖活动，代替普通的摇奖机，一样能做到合理、公正。在STC4.0软汉字系统支持下，我们编制了苹果电脑摇奖程序，有汉化的人机对话功能，操作十分方便。改变程序中有关软汉字系统的语句，该程序可适用于任何一种机型。整个程序简短、易读。

程序对每个开奖组的户数（用变量M表示）作了规定，要求能被1,000,000,000整除，这样中奖号码最多显示九个数字，一般八位微机都能直接胜任。为了使获奖机会均等，使中奖号码分布均匀，并使开出的中奖号码个数不至太多，程序还对每个奖级的中奖个数（用变量H表示）提出了能被户数M整除的要求，这和一般开奖的实际情况是相符的。该程序允许的开奖户数最多可达10亿，这在一个省市、甚至全国的范围内都是可行的。所设奖级个数（用变量N表示）则不作任何限制。

程序设计的关键，是确定某个奖级的中奖号码个数（用变量X表示）。一般地说，某个奖级有多少个奖，就有多少个中奖号码（X=H），当中奖个数较多时，令人眼花瞭乱。事实上，对奖是采用对编码末几位数字的方法，当中奖号码个数是10的整数倍时，应该把X尾部的若干个0去掉。例X=20时，只要开出二个中奖号码如470、970，而在其前分别添加0至9，就有0470、1470…9470，0970、1970…9970共二十个号码中奖。反之，为了使任何一组连续的奖券号都可用于户数为1.25×10^m、2×10^m、2.5×10^m、4×10^m、5×10^m或8×10^m等的开奖活动，有必要把户数虚拟地扩展为1×10^n（其最大可能是1×10^9），并把中奖个数H扩大同样的倍数，即$X=1\times10^9\times H/M$，并把可能产生的X尾部的0去掉。例如每25000户为一开奖组，设某等奖1个，则开奖号码个数$X=1\times10^9\times1/25000=4\times10^4$，去掉尾部的0，应开出中奖号码4个，即不仅在00000～24999间有一个号码，在25000～49000，50000～74999，75000～99999间各有一个中奖号码。程序的100和110号语句就是完成这项功能的。

程序的另一技巧，是确定中奖号码。根据计算机的特点，没有必要象普通摇奖机那样，一个一个数码地摇。具体的方法是：首先算出基本中奖号区间A，A是户数M除以中奖个数H的商，其次分别算出最大中奖号码的位数D和基本中奖号B（最初的B实际上也是最小的一个中奖号），再次，在进入打印中奖号码的循环后，算出中奖号前应添数码0的个数C，并打印出所有的中奖号码B（基本中奖号与区间A的值逐次累加）。计算C的步骤必不可少。例如当每10000户中有二个一等奖时，在基本号区间A=M/H=10000/2=5000中摇出基本中奖号753，算出最大中奖号码的位数D是4位，就必须在753前添加数码0得0753，否则按照对末几位数字的对奖方法，0753，1753，…，5753，…，9753，共10个号码都中奖，那是不行的。除了0753，另一个中奖号码是5753。

程序输入后可以试运行几次，确信其随机性后可打印输出。

```
10 INPUT"多少户为一开奖组？共设几个奖级？";M,N
20 Y=1E+9/M:IF Y<> INT (Y) THEN PRINT"户
   数不合要求，应能被1000000000整除，重新输入"：
   GOTO 10
30 INPUT"要打印吗？";P$: PRINT CHR$ (26)
40 IF P$ ="Y" THEN D$ = CHR$ (4):PRINT D
   $"STC B1":PRINT TAB(8)"苹果摇奖":PRINT
   M;"户为一开奖组，共设";N;"个奖级":PRINT D
   $"STC B0"
50 FOR I = 1 TO N
60 INPUT"几等奖？有多少个？";G,H
70 A=M/H: IF A<> INT (A) THEN PRINT"该等奖
   个数不合要求，应能被户数";M;"整除，重新输
   入": GOTO 60
80 IF P$ = "Y" THEN PRINT D$"STC B1"
90 PRINT G;"等奖，有";H;"个，中奖号码:": IF A=1
   THEN PRINT"所有号码":GOTO 200
100 X = Y * H
110 IF X/10= INT(X/10) THEN X=X/10:GOTO 110
120 D=LEN(STR$ (A * X-1)):B= INT (RND(1) *
```

高分辨率图形的趣味换页和清屏

苏州景范中学　赵旭

在编制微机辅助教学软件时，如果对高分辨率图形采用多种形式的换页和清屏，能增强软件的趣味性。这对激发学生的学习兴趣能起一定的作用。本文介绍二种实现高分辨率图形趣味换页和清屏的方法。

一种方法是借助于6502汇编语言。如程序一。能提供8种形式的换页或清屏。它们分别是：上下向中间；中间向上下；从上到下；从下到上；左右向中间；中间向左右；从左到右；从右到左。在BASIC程序中可用&H,L命令调用。其中L的值决定哪种形式(范围1～8)，而H的值为1是清屏，为2是换页。具体方法是先置&命令的入口地址(3F5:4C 27 61)。需趣味换页时，将图形放入第二页，并置第一页为操作页(POKE230,32)，用&命令调用后即能将第二页图形趣味地移到第一页。在趣味清屏时，如果POKE 230,32就清第一页，而POKE 230,64就清第二页。程序二为示范程序，将程序一调入内存后运行程序二，能将以上16种形式从头到尾表演一遍。表演后，如果觉得运行速度不合要求，可改动程序一中$6001单元中的值(范围0～255)。

另一种方法是在汇编程序基础上运用BASIC语言编出各种形式的换页和清屏程序。其优点是：程序容易理解，修改方便。

以下举例说明。程序三是将高分辨率局部图形在二页间相互传送。它有二个入口：$6000将第二页传送到第一页，而$6013将第一页传送到第二页。调用时，先将所传送的局部图形的左上角坐标X1、y1存入$00—$01单元，右下角坐标x2、y2存入$02—$03单元。并置操作页，当POKE 230,32时，则第二页传送到第一页。当POKE 230,64时，则第一页传送到第二页。

程序四是以程序三为基础，用BASIC语言编写的将高分辨率第二页的图形趣味传送到第一页的示范。共有4种形式：10行为水平交叉；20行为棋盘格显示；30行为中间向上下；40行为竖直交叉。如果第二页无图形即为将第一页趣味清屏。弄清了程序四后，就能根据自己的要求另外编出各种各样的换页与清屏的程序。

程序一

```
6000-  A9  90  20  A8  FC  8D  30  C0
6008-  60  20  0C  61  A0  00  A5  00
6010-  C9  01  D0  07  A9  00  91  06
6018-  4C  1E  60  20  F9  60  C8  C0
6020-  28  D0  EB  20  00  60  60  8A
6028-  48  20  09  60  86  FA  A9  BF
```

****

```
A)
130 FOR J=1 TO X
140 C=D- LEN (STR$ (B)):IF C =0 THEN 170
150 FOR K=1 TO C
160 PRINT 0;:NEXT K
170 PRINT B
180 B=B+A:NEXT J
190 IF P$ ="Y" THEN PRINT D$"STC B0"
200 NEXT I: END
RUN
多少户为一开奖组？共设几个奖级？1000,4 要打印吗？
Y
苹果摇奖
1000 户为一开奖组，共设 4 个奖级
1 等奖，有 1 个，中奖号码：
652
2 等奖，有 4 个，中奖号码：
068
318
568
818
3 等奖，有 20 个，中奖号码：
31
81
4 等奖，有 100 个，中奖号码：
5
```

```
6030-  38  E5  FA  AA  20  09  60  68
6038-  AA  60  98  48  20  FE  60  84
6040-  FA  A9  27  38  E5  FA  A8  20
6048-  FE  60  68  A8  60  20  E9  60
6050-  E8  E4  FF  D0  F8  60  A2  00
6058-  8A  48  20  11  F4  68  AA  A5
6060-  26  9D  00  94  A5  27  9D  00
6068-  95  E8  E0  C0  D0  EA  60  EA
6070-  20  56  60  A2  00  EA  20  27
6078-  60  E8  E0  60  D0  F7  60  EA
6080-  20  56  60  A2  5F  EA  20  27
6088-  60  CA  E0  FF  D0  F7  60  EA
6090-  20  56  60  A2  00  20  09  60
6098-  E8  E0  C0  D0  F8  60  EA  20
60A0-  56  60  A2  BF  20  09  60  CA
60AB-  E0  FF  D0  F8  60  EA  20  56
60B0-  60  A0  00  20  3A  60  C8  C0
60B8-  14  D0  F8  60  EA  20  56  60
60C0-  A0  13  20  3A  60  88  C0  FF
60CB-  D0  F8  60  EA  20  56  60  A0
60D0-  00  20  FE  60  C8  C0  28  D0
60D8-  F8  60  EA  20  56  60  A0  27
60E0-  20  FE  60  88  C0  FF  D0  F8
60E8-  60  20  0C  61  A5  00  C9  01
60F0-  D0  07  A9  00  91  06  4C  FD
60F8-  60  B1  08  91  06  60  A2  00
6100-  20  E9  60  E8  E0  C0  D0  F8
6108-  20  00  60  60  BD  00  94  85
6110-  06  85  08  BD  00  95  85  07
6118-  18  69  20  85  09  60  70  80
6120-  90  9F  AE  BD  CC  DB  EA  20
6128-  46  E7  A5  50  85  00  BD  1D
6130-  61  8D  35  61  4C  70  60  00
```

程序二

```
5 POKE 1014,39:POKE 1015,97:HGR2:POKE-16300,0:HCOLOR=3:FOR X=10 TO 270 STEP 5:HPLOT X,10 TO X,180:NEXT:FOR Y=10 TO 180 STEP 5:HPLOT 10,Y TO 270,Y:NEXT:POKE 230,32:CALL -3086:FOR L=1 TO 8:FOR H=2 TO 1 STEP -1:&H,L:FOR T=1 TO 500:NEXT T,H,L
```

程序三

```
6000-  A6  01  20  48  60  18  69  20
6008-  85  09  20  56  60  E8  E4  03
6010-  D0  F0  60  A6  01  20  48  60
6018-  38  E9  20  85  09  20  56  60
6020-  E8  E4  03  D0  F0  60  8A  29
6028-  C0  85  26  4A  4A  05  26  85
6030-  26  8A  85  27  0A  0A  0A  26
6038-  27  0A  26  27  0A  66  26  A5
6040-  27  29  1F  05  E6  85  27  60
6048-  20  26  60  A5  26  85  06  85
6050-  08  A5  27  85  07  60  A4  00
6058-  B1  08  91  06  C8  C4  02  D0
6060-  F7  60  00  00  00  00  00  00
```

程序四

```
5 HGR2:HGR:POKE -16302,0:POKE 230,64:HCOLOR=3:FOR X=10 TO 279 STEP 5:HPLOT X,10 TO X,180:NEXT:FOR Y=10 TO 180 STEP 5:HPLOT 10,Y TO 270,Y:NEXT:POKE 230,32
10 FOR I=0 TO 38:J=38-I:POKE 0,I:POKE 1,0:POKE 2,I+1:POKE 3,47:ALL 24576:POKE 0,J:POKE 1,47:POKE 2,J+1:POKE 3,95:CALL 24576:POKE 0,J:POKE 1,143:POKE 2,J+1:POKE 3,191:CALL 24576:POKE 0,I:POKE 1,95:POKE 2,I+1:POKE 3,143:CALL 24576:NEXT
15 FOR T=1 TO 600:EXT:CALL -3086
20 FOR Y=3 TO 141 STEP 46:FOR X=0 TO 33 STEP 6:POKE 0,X:POKE 1,Y:POKE 2,X+3:POKE 3,Y+23:CALL 24576:POKE 0,X+3:POKE 1,Y+23:POKE 2,X+6:POKE 3,Y+46:CALL 24576:FOR T=1 TO 100:NEXT T,X,Y:POKE 0,0:POKE 1,0:POKE 2,39:POKE 3,191:FOR T=1 TO 400:NEXT :CALL 24576
25 FOR T=1 TO 600:NEXT:CALL -3086
30 FOR I=90 TO 0 STEP -10:J=190 -I:POKE 0,0:POKE 1,I:POKE 2,39:POKE 3,J:CALL 24576:NEXT
35 FOR T=1 TO 600:NEXT:CALL -3086
40 FOR I=0 TO 180 STEP 10:J=180-I:POKE 0,0:POKE 1,I:POKE 2,10:POKE 3,I+10:CALL 24576:POKE 0,20:POKE 1,I:POKE 2,30:POKE 3,I+10:CALL 24576:POKE 0,10:POKE 1,J:POKE 2,20:POKE 3,J+10:CALL 24576:POKE 0,30:POKE 1,J:POKE 2,39:POKE 3,J+10:CALL 24576:NEXT
```



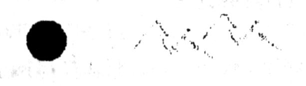
黑龙江省呼兰师专　　吴全有

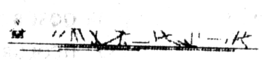
本程序是利用图形表方法编写的划船动画。背景为山、亭和太阳。草木是随机产生的。开机后，自动划船，前进方向随机产生，船桨不断摆动。为增加动画效果，加入了划桨的声音，使之如临其境。船在画面水域内不断改变位置，在临近边界时自动调整方位，运行起来很有趣味。该程序在 APPLE Ⅱ 机上调试通过。

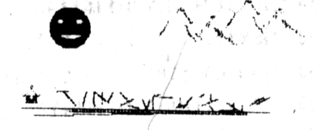
附程序清单及画面。

```
10  HGR2:HCOLOR=3:HOME
20  X= 80:Y0 =30:A=15
30  FOR I=H. 5 * 3. 14159 TO 1. 5  * 3. 14159 STEP 1/A
40  X=1. 1  * A * COS (I):Y=A * SIN(I)
50  H1=X0+X:V=Y0-Y:H2=X0+ABS (X)
60  HPLOT H1,V TO H2,V
70  NEXT I
80  GOSUB 850
90  FOR N=0 TO 150
100  READ T:POKE 24576+N,T
110  NEXT N
120  FOR N=0 TO 193
130  READ T:POKE 24750+N,T
140  NEXT
150  FOR N=0 TO 185
160  READ T:POKE 24950+N,T
170  NEXT N
180  POKE 232,0:POKE 233,96:ROT=0:SCALE=1
190  DRAW 1 AT 40,80:DRAW 2 AT 153,35
200  DRAW 2 AT 210,28
210  GOSUB 950
220  H=30:V=190
230  FOR 1=0 TO 200 STEP 20
240  HPLOT H+I,V TO H+I+30,V-4
250  NEXT
260  H=50:V=90
270  HPLOT H-10,V TO 250,V
280  HPLOT H,V+5 TO 150,V+2
290  HPLOT 80,V+2 TO 230,V+1
300  H=140:V=140:T=0
305  R=RND (1)
310  IF R < 0. 5 THEN A $ ="L"
315  IF R > =0. 5 THEN A $ ="R"
320  IF A $ ="L" THEN 350
330  IF A $ ="R" THEN 390
340  PRINT CHR $ (7):GOTO 310
350  H=H-5:V=V-2:T=T+1
360  IF H < 40 THEN A $ ="R":GOSUB 640:GOTO 320
370  IF V < 100 THEN 770
380  GOSUB 510:GOTO 320
390  H=H+5:V=V+2:T=T+1
400  IF H > 230 THEN A $ ="L":GOSUB 510:GOTO 320
410  IF V > 150 THEN 810
420  GOSUB 640:GOTO 320
430  HPLOT 70,24 TO 73,24:HPLOT 86,24 TO 89,24
440  HPLOT 69,25 TO 74,25:HPLOT 85,25 TO 90,25
460  P=70:Q=35
470  FOR N=0 TO 4
480  HPLOT P+N * 3,Q+N TO 90 - N * 3,Q+N
490  NEXT N
500  RETURN
510  POKE 232,174:POKE 233,96
520  HCOLOR=3:DRAW 1 AT H,V
530  DRAW 2 AT H,V
540  FOR N=1 TO 100:NEXT
550  HCOLOR=0:DRAW 2 AT H,V
560  HCOLOR=3:DRAW 3 AT H,V
570  FOR N=1 TO 200:NEXT
580  GOSUB 900
590  IF T/2= INT (T/2) THEN HCOLOR=0:GOSUB 430:
     GOSUB 610
600  HCOLOR=1:GOSUB 430
610  POKE 232,174:POKE 233,96
620  HCOLOR=0: DRAW 1 AT H,V:DRAW 3 AT H,V
630  RETURN
640  POKE 232,118: POKE 233,97
650  HCOLOR=3: DRAW 1 AT H,V
660  DRAW 2 AT H,V
670  FOR N=1 TO 50:NEXT
680  HCOLOR= 0: DRAW 2 AT H,V
690  HCOLOR= 3: DRAW 3 AT H,V
700  FOR N=1 TO 200: NEXT
710  GOSUB 900
720  IF T/1= INT (T/2) THEN HCOLOR=0: GOSUB 430:
     GOSUB 740
730  HCOLOR=3: GOSUB 430
740  POKE 232,118: POKE 233,97
750  HCOLOR= 0: DRAW 1 AT H,V: DRAW 3 AT H,V
760  RETURN
770  H=H-5:V=V+2:T=T+1
780  IF H<40 THEN A $ ="R":GOTO 320
```

```
790  IF V > 150 THEN 350
800  GOSUB 510: GOTO 770
810  H=H+5:V=V-2:T=T+1
820  IF H > 230 THEN A$="L": GOTO 320
830  IF V < 100 THEN 390
840  GOSUB 640: GOTO 810
850  FOR I=770 TO 790
860  READ N
870  POKE I,N
880  NEXT I
890  RETURN
900  FOR I=250 TO 230 STEP - 5
910  POKE 768,I: POKE 769,30
920  CALL 770
930  NEXT I
940  RETURN
950  DIM H(30),V(30)
960  FOR I=1 TO 30
970  H(I)=INT ( RND (9) * 18):V (I)=INT (RND (3)
     * 15)
980  NEXT I
990  H=60:V=90
1000  FOR I=1 TO 29
1010  H=H+6
1020  HPLOT H+H(I),V-V(I) TO H+H(I+1),V-V(I+1)
1030  NEXT I
1040  RETURN
1050  DATA 173,48,192,136,208,5,206,1,3,240,9,202,
      208,245,174,0,3,76,2,3,96
1060  DATA 2,0,6,0,51,0
1070  DATA 14,101,12,12,12,12,4,96,21,30,118,14,
      14,14,77,30,63,63,255,63,55,45,77,41,53,63,
      223,59,55,45,45,45,45,254,59,255,187,45,45,
      45,45,45,5,0,0
1080  DATA 12,12,44,44,44,36,101,100,36,13,12,12,
      12,12,12,12,100,44,12,54,101,46,22,53,46,46,
      46,46,110,62,255,27,22,53,53,54,53,46,46,46,
      46
1090  DATA 53,46,110,73,1,224,60,56,231,60,109,97,
      36,101,12,12,100,12,100,12,44,13,62,191,169,
      13,30,223,179,55,182,79,73
1100  DATA 40,109,62,55,13,109,254,191,9,109,254,
      59,23,77,41,13,46,109,62,87,109,190,9,109,62,
      0
1110  DATA 3,0,8,0,141,0,166,0
1120  DATA 30,63,224,28,28,23,23,23,23,63,176,74,
      9,12,109,62,63,55,45,45,62,63,55,45,45,62,63,
      55,45,45,30,63,46,173,63,63,23,45,45,45,62,
      63,63,55,45,45,45,30,63,55,45,45,21,63,63,46,
      45,53,63,63,159
1130  DATA 45,45,45,30,63,31,63,23,45,77,45,21,63,
      223,59,55,109,73,173,63,223,59,46,77,9,45,21,
      63,223,27,255,219,27,63,39,151,9,45,45,45,45,
      45,45,45,45,45,45,45,172,218
1140  DATA 63,63,63,63,63,63,63,63,63,63,14,45,45,
      45,45,45,45,45,45,45,5,0
1150  DATA 150,146,146,18,191,215,63,63,246,4,0,
      32,188,18,30,46,30,30,191,191,191,191,191,63,
      0
1160  DATA 150,146,146,18,30,30,30,28,63,63,7,32,
      188,50,54,30,54,54,24,30,30,62,62,190,23,23,
      7,0
1170  DATA 3,0,8,0,142,0,161,0
1180  DATA 14,45,96,12,12,21,21,21,21,45,176,219,
      216,63,23,45,45,62,63,55,45,45,62,63,119,53,
      63,23,45,45,21,63,63,63,46,45,45,53,63,63,
      119,45,45,62,63,191,45,45,62,63,55,45,45
1190  DATA 62,63,55,45,45,21,63,63,63,14,45,13,45,
      21,63,31,63,46,109,45,62,255,59,191,45,77,41,
      62,255,219,63,46,109,73,45,245,219,216,219
1200  DATA 63,23,109,73,41,109,73,73,9,30,63,63,
      63,63,63,63,63,63,63,63,63,176,9,45,45,45,45,
      45,45,45,45,30,63,63,63,63,63,63,63,31,0,0
1210  DATA 150,146,146,42,45,45,45,45,11,21,31,
      191,73,117,174,117,14,46,0
1220  DATA 150,146,146,170,21,13,45,45,13,32,172,
      210,14,54,54,54,46,21,21,21,46,46,46,46,0
```







·趣味程序·

计算费马数 $F_{(n)}=2^{2^n}+1$

江苏省常熟中学　薛维明

数学家在寻找能计算出素数公式的漫长的道路上，进行了不懈的努力。他们中间有失败的教训和启迪，也有成功的喜悦。当然，在这当中还有不少有趣的插曲。

1640年法国数学家费马(Fermat，1601——1665年)，曾认为他找到了能算出素数的公式，这就是有名的费马数 $F_{(n)}=2^{2^n}+1$。

因为 $F_{(0)}=2^{2^0}+1=3$

$_{(1)}=2^{2^1}+1=5$

$F_{(2)}=2^{2^2+1}=17$

$F_{(3)}=2^{2^3}+1=257$

$F_{(4)}=2^{2^4}+1=65537$

这些数都是素数。当时，费马没有作进一步的计算和进行证明。但他坚信所有这种数都是素数，并确信无疑。

然而，在费马宣布这个公式的一个世纪后。数学家欧拉(Euler，1707—1783年)，在1731年计算出 $F_{(5)}=4294967297$，可被641整除。即

$4294967297=6700417\times641$

到1971年布里哈特(John Brillhart)和莫里森(Michael Morrison)，利用加利福尼亚大学洛杉机分校的一名计算机找到了 $F_{(7)}$ 的两个素因子

$F_{(7)}=2^{2^7}+1=2^{128}+1=340282366920938463463374607431768211457=59649589127497217\times5704689200685129054721$

从目前的资料来看，当 $n\geqslant5$ 费马数多不是素数。要判别一个较大的费马数是否是素数，确实不简单。要计算出一个费马数也不容易。现在用BASIC语言编写一个能计算出费马数的程序，在CEC—I机上仅运行十一分三十二秒钟，就能计算出F=0至8的费马数，并能高精度打印出来。

程序如下(运行法省略)：

```
20  DIM A(100)
22  PRINT "N=";
23  INPUT N
25  PRINT : PRINT
26  PRINT "N=0";" F(0)=3"
27  PRINT "N=1";" F(1)=5"
30  FOR M=2 TO N
40  LET L=INT (2^ M+0.1)
45  PRINT "N=";M;" F(";M;")=";
50  LET Q=2
60  LET D=INT (L* LOG (Q)/LOG (10)) +2
80  LET K=1
90  LET A(1)=Q
100  FOR I=1 TO L-1
110  LET B=0
115  FOR J=1 TO K
120  LET C=A(J) *Q
140  LET A(J)=C-INT (C/10) *10+B
150  LET B=INT (C/10)=INT (A(J)/10)
160  LET A(J)=A(J)-INT (A(J)/10) *10
170  NEXT J
175  LET A(J)=B
180  IF A(J) > 0 THEN K=K+1
190  NEXT I
200  FOR J=K TO 2 STEP -1
210  PRINT A(J);
220  NEXT J
230  PRINT A(1)+1
235  NEXT M
265 END
```

更　正

期	页	行	误	正
一	3	倒13	教师	教学
一	5	3	颢	灏
一	11	左栏倒2	度	序
二	6	右栏倒4	$F	$7F
二	11	左栏倒12	$DAF	$8DAF
二	17	左栏倒12	胶,280	们,Z80
二	11	左栏倒22	删去"每次……26种组合键"	

SD-50Mark的主轴恒速驱动电路

上海机械学院计算中心　彭大一

SD-50Mark Ⅱ 型软磁盘驱动器属 5.25 英寸、单面(一个读/写磁头)软磁盘驱动器。许多 Apple 微机均配置该驱动器。微机主机通过接口卡和软盘控制器实现对它的控制。从而达到主机与外部存储器(软磁盘)间的数据交换。

对于 5.25 英寸软磁盘驱动器,其主轴转速必须控制在(300±0.5)转/分,该转速必须均匀稳定,不随负载的变化和电源的波动而变化,这是使驱动器对软磁盘正确地读/写数据信号的必要条件之一。因此,有关电路均采用闭环调速方式来达到恒速驱动。

SD-50Mark Ⅱ 型软磁盘驱动器中,驱动软磁盘转动的电机为直流无刷电机,其恒速驱动电路装配在一块独立的印刷线路板上,该板的俯视图如图一所示。在实际运用中,该恒速驱动器比较容易出故障,故障现象为电机不转或转速不正常(调谐 VR1 无效果),若板上的 TR1 和 TR2 二只三极管没有损坏,则极可能是集成芯片 LAG570 损坏了。由于 LAG570 目前在国内市场上几乎买不到,因此,该元件的损坏就造成整台驱动器不能使用。为解决这一问题,特向用户提供另一驱动器主轴恒速驱动电路,其线路如图二所示。该电路简单,成本不高,将其安装在与原线路板同样尺寸的线路板上,便可固定在驱动器内。其恒速驱动效果很好。下面对其工作原理作简单介绍:

一、当主机不对软磁盘进行读/写时,软盘控制器送出的 MOTORON 信号为高电平,经电阻分压,使 Q_2 基板为高电平,Q_2 饱和导通,因此,$V_{Q2C} \approx V_{Q2E} = 0V$(接地)。又由于 $V_{Q1B} = V_{Q2C}$,因此,Q_1 截止,电机不转。

二、当主机要进行读/写时,软盘控制器送出的 MOTOR ON 信号为低电平。因此,Q_2 截止。此时,由于电机未转,LM2917 的输入端为零,LM2917 的比较器的同相输入端为零,其反相输入端接正电平(由 R_4 分压),这导致 LM2917 中的功放由于基板为负电位而截止。由图二可知,此时,+12V 直流电压经 R_8 分压,建立了 Q_1 的基极电压,致使 Q_1 导通,电机开始转动。

电机转动后,电机中的测速绕组输出与电机转速成正比的信号送 LM2917 的输入端(1 脚和 11 脚),经放大、整形以及电荷泵转换成与电机转速成正比的直流电压送到比较器的同相输入端,比较器将该信号与其反相输入端的输入电压比较,当 $V_{同相} > V_{反相}$,并且达到某一正值时,LM2917 中的功放导通,形成由 +12V →R_8→LM2917 中功放的 C、E 极→R_9→地的回路。可知,LM2917 中功放的集电极电压 $V_C = V_{Q1B}$,因此,LM2917 中功放的导通将使 V_{Q1B} 降低,则 Q_1 的 C、E 间内阻增大,致使电机转速下降,也就是说,LM2917 功放的导通程度直接控制 Q_1 的导通程度,进而控制电机的转速。

如果通过 R_4 给定一个正电压,则电荷泵输出的电压与其比较,并且送出一个差值电压来控制 LM2917 中功放的导通程度,就可以使电机保持某一恒定转速。

当由于某种原因(如加载)造成电机转速下降时,电机测速绕组的输出经电荷泵转换后输出的直流电压相应下降,而比较器的反相输入端电压不变(由 R_4 给定),因此,比较器输出电压降低,LM2917 中功放的导通程度下降,其集电极电位上升,则 V_{Q1B} 电位上升,Q_1 导通增强,电机的转速提高。

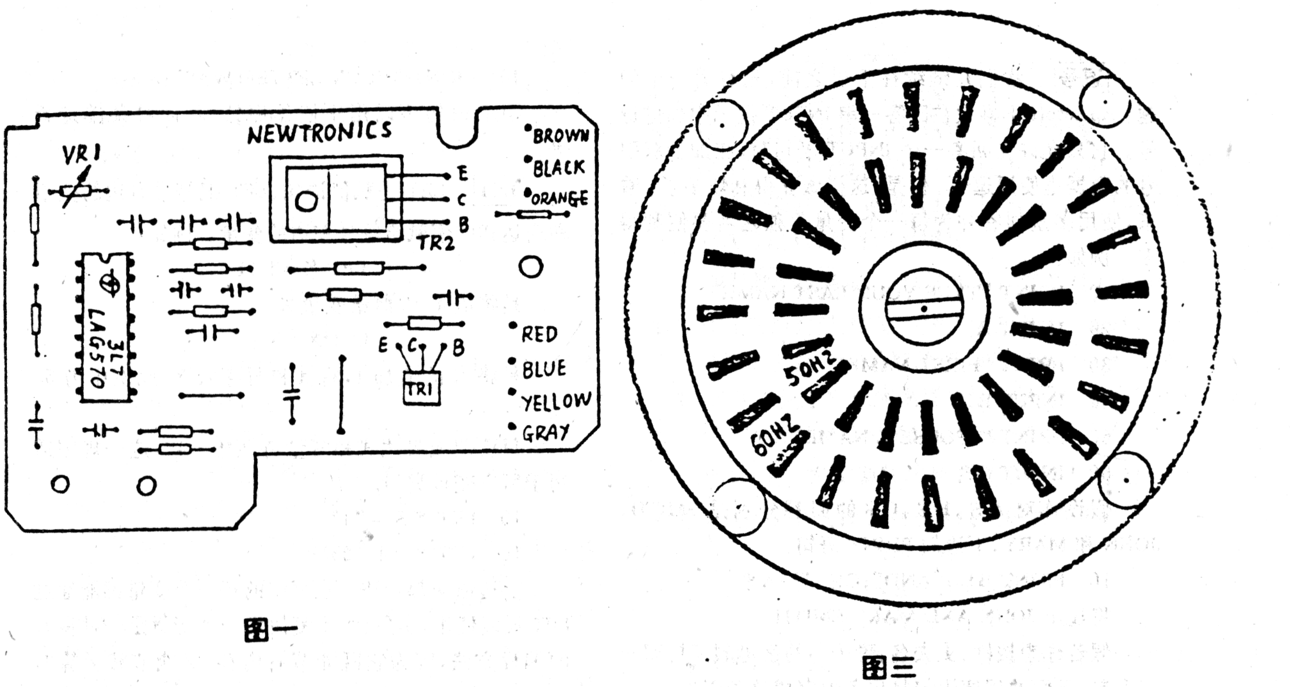
反之,当电机转速大于给定转速时,比较器输出电压上升,LM2917 中功放的导通程度增强,其集电极电位下降,则 V_{Q1B} 电位下降,Q_1 导通减弱,电机的转速降低。

综上所述,可知,电机的某一恒定转速由 R_4 给定。在实际运行中,Q_1 的导通程度随电机转速的微弱变化而变化,不断地进行调整,使驱动器的主轴转速稳定在 300 转/分。

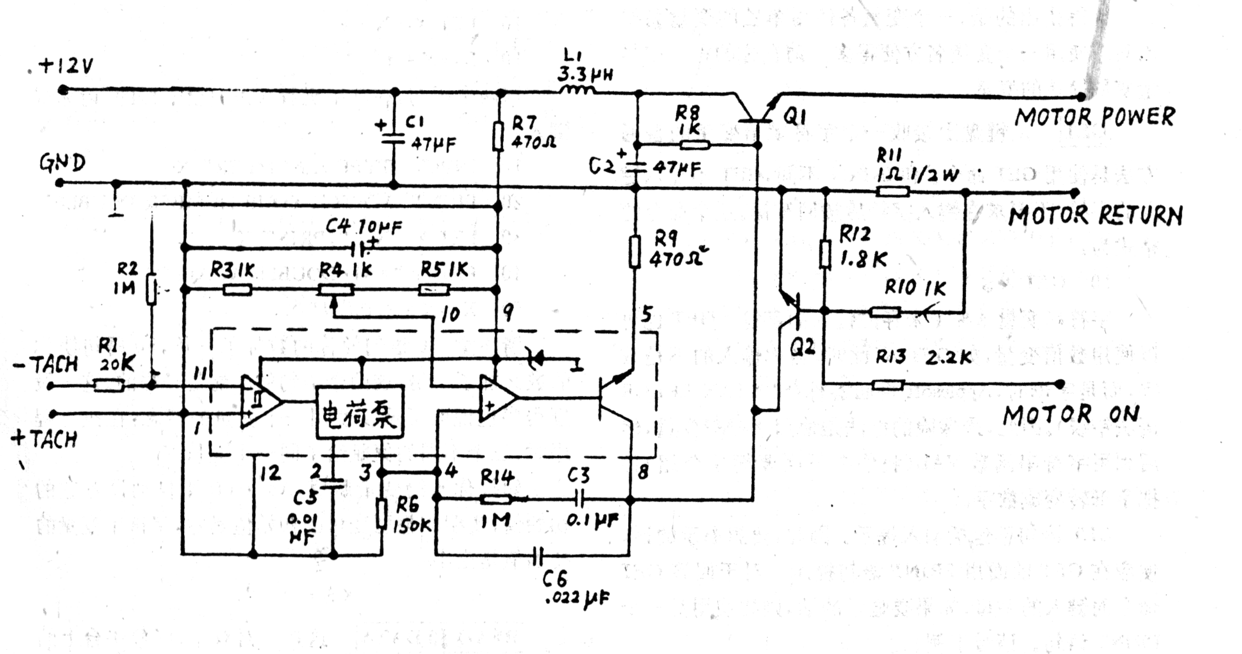
装上了如图二所示的恒速驱动电路后,在调整驱动器主轴转速时,一面用螺丝刀慢慢地调谐 R_4,一面观察驱动器背面的主轴飞轮上贴着的转速测试卡(该卡如图三所示),该卡内环上标有 50HZ 字样。必须将 R_4 调到内环出现稳定的测试卡图案为止,此时,驱动器的主轴转速为额定的 300 转/分。如果出现图案转动或看不清图案,均为主轴转速不对。

原 SD-50Mark Ⅱ 软磁盘驱动器恒速驱动电路有七根引出线,现将其说明如下:

红、兰线——电机绕组引线　　　　　　　　桔线——MOTOR ON 引线
黄、灰线——测速绕组引线　　　　　　　　黑线——地线
棕线——＋12V 直流电源线



图一　　　　　　　　　　　　　　　　　　图三



图二　注：图中电容耐压值均为 20V 或 20V 以上

APPLESOFT BASIC 入门(连载)

逗号 我们在继续往下讲之前，先看另一个问题。Applesoft 不喜欢在回答一串 INPUT 提示中加进逗号。它只允许在回答一个 INPUT 语句时，在加了双引号的数据里使用逗号。但是，这样做显得麻烦多了。看来，使用多重回答并为每一个变量单独起名要便利得多。例如：

```
10  PRINT "TYPE YOUR LAST NAME"
20  INPUT A
30  PRINT "FIRST NAME"
40  INPUT B$
50  PRINT "SPOUSE'S NAME"
60  INPUT C$
```

假设你对 A$，B$，C$ 的回答分别是 SMITH，JOHN 和 MARY，下边的 PRINT 语句：

```
10  PRINT B$;"AND";C$;"";A$
```

将显示 JOHN AND MARY SMITH

假若你想按姓、丈夫名、妻子名的字母顺序打印一张名单，下面的程序语句将出色地完成这项工作：

```
10  PRINT A$;". ";B$;"&";C$
```

值得指出的是，三个变量各使用单独的变量名要比共同使用一个变量名方便得多。请在你的程序中尽量使用简单的变量。

GET 从键盘上接收一个字符赋给变量的普通方法是使用 GET 命令。和 INPUT 不同，GET 不显示键入的字符，不要求在键入字符后按回车键。这个命令的格式为：

```
10  GET A$
```

字符串变量 A$ 用来存贮键入的字符。GET 也可以使用数值变量，如 GET A，此时，当你输入的不是数字，而是字母时，Applesoft 将回答你(SYNTAX ERROR 语法错误)。因此，最保险的办法是使用字符串变量，然后利用字符串函数 VAL(我们在下面要简单介绍它)把字母转变成数字。

GET 语句不包括输入提示。因此，要想提示的话，便应在 GET 前面用 PRINT 语句提示。对于回答 GET 命令而键入的字母，如果要显示的话，则要使用另一个 PRINT 语句。请看下例：

```
10  PRINT "TYPE ANY LETTER";
20  GET X$
40  PRINT "YOUR LETTER WAS";X$
```

GET 语句可以用来使程序暂停，回答菜单选择等。

LET LET 是赋值语句，LET 把等号右边的变量和表达式的值赋给等号左边的变量。例如：

LET X=4.75

是将值 4.75 赋给变量 X

LET X=X+1

是将 X 的值加 1，将其和再存入 X，此例中为 5.75。

只要值或表达式和变量特性相一致，这一规则适用于所有变量类型。

```
10  LET X$="4"
10  LET X=4.12359
```

上边语句都合法。程序中的每一个变量都能通过 LET 反复使用，实际上，反复使用一个变量是一种良好的编程方法，因为它既能节省内存，又能节省运算时间。

LET 也可以省略，换句话说，

```
10  LET X=X+4
10  X=X+4
```

这两个语句，在苹果机看来是完全一样的。请你试键入：

```
10  INPUT "TYPE A NUMBER";X
20  PRINT "WATCH YOUR NUMBER DOUBLE"
30  LET X=X*2:PRINT X
40  PRINT "AND DOUBLE AGAIN"
50  X=X*2:PRINT X
```

请注意，30 语句使你的数加了一倍，50 语句使你的数又加了一倍，这两个语句的功能是一样的。这一段短程序也说明了反复使用一个变量的概念：在执行程序时，三个不同的值赋给了同一个变量 X。

如果你不给变量赋值，Applesoft 则自动设置它的初始值，实型与整型变量的初始值是零，字符串变量的初始值是空串：

X$=" "

READ 和 DATA 这是一对双生，好象午台上的双簧一样，二者必须配合使用，缺一不可。

DATA 语句，可以建立一张数据表，然后供 READ

语句来使用。例如,下面是一个含有五个数字的DATA语句,数字之间由逗号隔开:

```
10 DATA 3.14,2,86.2,39.1,17
```

你只要记住以下几条简单的规则,便可很容易地编写DATA语句:

1、DATA语句可在程序中的任何位置使用,可以累积。也就是说,假若你的程序中有五个数据项,你可以用上面列举的一行DATA语句,也可以建立五个独立的DATA语句:

```
10 DATA 3.14
20 DATA 2
30 DATA 86.2
40 DATA 39.1
50 DATA 17
```

2、要用READ语句读入数值变量(整型或实型)的数据必须是数字值。

3. 任何类型的数据均可读入字符串变量。

4. 假若把字符串读入数值变量,Applesoft在执行到这里时将出现语法错误信息。

用DATA语句建立数据表后,READ可把表中的数赋值给READ语句中的变量。第一个变量取DATA数据表中的第一个数值,第二个变量取第二个数值,依此类推。当Applesoft遇到第一个READ语句时,它就把DATA数据表中的第一个数据项赋予第一个READ语句中的变量。遇到第二个READ语句时,就从第二个数据项取数,赋予第二个READ的变量。Applesoft对于使用多少个数据项是严格跟踪查看的。如果程序中的READ语句读取的数据超过DATA所含有的数据,电脑就会显示超出数据范围的信息"OUT OF DATA ERROR IN"。

象教骑自行车一样,讲清READ和DATA语句比实际使用它们难。还是让我们编一段程序来说明它们吧:

```
10 READ A$,B$,C$
20 PRINT A$,B$,C$
30 READ D
40 PRINT D
50 READ E$
60 PRINT E$
70 DATA "USING","READ&","DATA"
80 DATA 789.12,355,"THE END"
90 READ F$
100 PRINT F$
```

你可以看到,程序里使用了六个数据项,依次将它们赋值给规定的变量。虽然我们把DATA语句放到了某些READ语句的前面及某些语句的后面,但READ语句总能依次找到数据表中的数据。

RESTORE 在执行READ和DATA语句时,你可用RESTORE命令随时返回数据表的初始点。它后面不要任何参数、条件或选择项,只要键入:

```
10 RESTORE
```

即可返回DATA数据表首。以后遇到READ语句时,它就将读取DATA表的第一项数据。这个命令使你能反复使用你程序中的DATA语句的数据。

这些是给变量赋值的最主要的方法。你可看到,变量对你编程来说十分重要,因此,应当反复练习本讲中介绍的这些语句,熟练地掌握它们。

变量类型的转换 我们前面已经学过,Applesoft的变量有三种类型:整型、实型、字符串型。有时,在程序中可能需要把一种类型的变量转变成另一种类型。例如,把实型变量转换成整型变量,或转换成字符串等等。有一些语句便是用来转换变量类型的。

INT(n) 把实型数转换成整形数。它是把一个数的小数点部分去掉来完成这项工作的。例如:

```
10 X=3.75
20 X$=INT(X)
30 PRINT X$
```

你看是不是显示整型数了?INT函数对变量值不进行四舍五入,而是直接去掉小数点部分,显示不大于n的最大整数。若要想使n四舍五入,在使用INT函数前加上0.5即可:

```
10 X=3.75:Y=3.5:Z=3.1
20 X$=INT(X+0.5)
30 Y$=INT(Y+0.5)
40 Z$=INT(Z+0.5)
50 PRINT X;"IS ROUNDED TO";X%
60 PRINT Y;"IS ROUNDED TO";Y%
70 PRINT Z;"IS ROUNDED TO";Z%
```

这时,每一个变量都作了四舍五入。

相反,从整型数转换成实型数,使用LET语句:

```
10 LET X=X%
```

或者

```
10 X=X%
```

VAL(X$) 把字符串转换成实型或整型变量。格式是:

```
10 X=VAL(X$)
```

或者

```
10 X%=VAL(X$)
```

字符串X$中第一个字符应为数字,否则整个字符串将转变为零。VAL(X$)首先查看字符串表达式

第一个字符是不是数字，若不是，将它的值取为零。例如：

```
X$="3.1416"
X=VAL(X$)
PRINT X
```

这时显示X的值为3.1416。如果把X$变成下面的情况，则会得到不同的结果：

```
X$="P1=3.1416"
X=VAL(X$)
PRINT X
```

这时X的值则为零。而下面的情况，结果也不一样：

```
X$="3POINT1416"
X=VAL(X$)
PRINT X
```

这时X=3。只要仔细地检查要转换的字符串表达式，就会避免这些差错。

在你练习编程时，如果对输入数据的语句误输入字符串，屏幕出现REENTER的出错信息。为了避免这一点，一般都是用字符串变量来回答输入语句，然后用VAL进行转换。例如：

```
10  INPUT"ENTER A NUMBER:";X$
20  N=VAL(X$)
30  PRINT N
```

STR$(X) 将数值变量转换成字符串变量。要记住，转换的目的变量是字符串，格式为：

```
10  X$=STR$(X)
```

这一语句对显示数字整齐、小数点对齐十分有用。

字符串处理 字符串处理是一种有用的实用技能，它使你从一个字符串中抽取特定的字符，称为“子串”。弄清楚字符串处理的最好方法便是在实际中使用它。

LEFT $(X$,n) 取字符串X$中最左边的n个字符。例如，若X$="PROGRAMMING"，那么：

```
PRINT LEFT$(X$,3)
```

显示出PRO。

假若n的数值比表达式中的字符数多，则只取出字符串实有的字符，多余的位删掉。当然，也可以把取出的子串赋值给程序中的另一个变量：

```
10  X$="PROGRAMMING"
20  Y$=LEFT$(X$,4)
30  PRINT Y$
```

这时，屏幕显示PROG。LEFT$也能处理用双引号括起来的字符串，如：

```
20  Y$=LEFT$("PROGRAMMING",4)
30  PRINT Y$
```

RIGHT(X$,n) 取字符串X$中最右边的几个字符。我们仍用上边的同一字符串X$：

```
PRINT RIGHT $(X$,4)
```

这时显示：MING。也可在程序中直接从字符串取字符赋值给另一变量。格式如下：

```
10  Y$=RIGHT$("PROGRAMMING",3)
20  PRINT Y$
```

屏幕上将显示ING。

MID$(X$,n) 从字符串当中抽取字符。如果使用下边的命令格式：

```
MID$(X$,n)
```

则取出的子串是从X$字符串第n个字符开始直到结尾所有的字符。例如：

```
PRINT MID$("PROGRAMMING",4)
```

屏幕显示：GRAMMING

如果使用的是字符串变量X$，结果相同。

另外，还有另一种确定取字符的方法：

```
MID$(X$,n,m)
```

这时，是从字符串X$的第n个字符起向右取m个字符。

```
PRINT MID$("PROGRAMMING",4,3)
```

想一想，会出现什么情况？是不是显示GRA三个字符？

以上三个语句初看起来虽然很琐碎，但却有很大实用价值。请你自己试编几个字符串表达式，练习从字符串中抽取特定的字符，以熟练掌握这几个语句的用法。

LEN(X$) 用来测定字符串字符的个数。如果我们仍使用上面的字符串例子(X$="PROGRAMMING")，那么：

```
PRINT LEN(X$)
```

则显示11。我们也可以使用字符串表达式：

```
PRINT LEN("PROGRAMMING")
```

得出相同的结果。你也可把得出的结果赋值给程序中的一个数值变量：

```
10  X$="PROGRAMMING"
20  X=LEN(X$)
30  PRINT X
```

结果是X等于11。LEN语句在屏幕显示格式或数字行对齐方面特别有用。

字符串连接 在两个或多个字符串之间用加号(+)连接起来，实现字符串的合并。请看下面一段短程序：

```
10  X$="PRO"
20  Y$="GRAMM"
30  Z$="ING"
40  PRINT X$+Y$+Z$
```

屏幕上将出现:PROGRAMMING。

数组 整型、实型、字符串型变量每一种都可以排列成一张表,称为数组。这种表共用相同的变量名与变量类型,但使用一种数字指示,指明表中使用哪一个元素,这种数字指示称为下标。下标可以是正整数或代表正整数的变量,用园括号括起来,位于变量名后面。例如:

```
X$(1)="PROGRAMMING"
```

下面一段短程序说明数组变量的一种使用方法:

```
10  X$(1)="PROGRAMMING"
20  X$(2)="THE APPLE"
30  X$(3)="IS FUN"
40  INPUT "WHICH STRING? (1-3)";N
50  PRINT X$(N)
```

第10—30行,程序,把每一个字符串赋值给各自的数组元素。在40行,要求用户回答一个数,并把这个数赋给变量N。在50行,变量N用来作为数组元素的下标。

到现在为止,我们介绍过的变量都属于简单变量。在运行程序时,Applesoft自动为各简单变量向内存空间赋值。它自动为数组变量最多赋值11个元素(下标0到10),如果数组元素超过11个时,则要求有足够大的内存空间,这时要使用DIM语句,即在数组变量执行之前,按下面的格式规定数组的大小:

```
DIM X$(100)
```

这一语句是为数组元素准备100个房间(从0到100,实际上为使程序更简洁,0常略去)。

数组元素也可以有两维,这是多维数组,本文不准备谈它的使用。

变量程序 LISTING1中的程序演示了本讲中介绍过的一些新命令。你可以注意到它用了许多上一讲介绍过的PRINT命令。请将这个程序键入你的电脑,并运行它,实际验证变量的各种用法。

```
LISTING 1:VARIABLES
10  REM VARIABLES
20  TEXT:HOME:NORMAL
30  VTAB 7:PRINT "THIS PROGRAM DEMONSTRATES
    THE TECHNIQUES":PRINT "DESCRIBED IN THIS
    ARTICLE。"
40  GOSUB 260
50  INVERSE:PRINT "THE INPUT STATEMENT:NORMAL
60  VTAB 7:INPUT "ENTER YOUR FIRST NAME:";A$
70  VTAB 8:INPUT "ENTER YOUR LAST NAME:";B$
80  VTAB 11:PRINT "YOUR NAME IS:";A$"  ";B$
90  GOSUB 260
100 INVERSE:PRINT "LEFT$,RIGHT$,AND MID$":
    NORMAL
110 A$=A$+B$
120 VTAB 7:PRINT "THE FIRST 3 LETTERS OF YOUR
    NAME:";LEFT$(A$,3)
130 VTAB 10:PRINT "THE LAST 3 LETTERS OF YOUR
    NAME:";RIGHT$(A$,3)
140 VTAB 13:PRINT "THE MIDDLE 3,LETTERS OF YOUR
    NAME:";MID$(A$,LEN(A$)/2,3)
150 GOSUB 260
160 INVERSE:PRINT "READ AND DATA":NORMAL
170 DIM N$(10):VTAB 5:PRINT "THE FOLLOWING
    WORDS HAVE BEEN READ INTO":PRINT "THE
    ARRAY N$() FROM DATA STATEMENTS":PRINT
    "CONTAINED IN THIS PROGRAM:":PRINT
180 FOR X=1 TO 10:READ N$(X):PRINT N$(X):
    NEXT
190 GOSUB 260
200 INVERSE:PRINT "ARRAYS":NORMAL
210 VTAB 5:PRINT "HERE IS THE CONTENTS OF THE
    ARRAY N$()":PRINT READ "FROM DATA
    STATEMENTS IN THE LAST":PRINT "SECTION:":
    PRINT
220 FOR X=1 TO 10:PRINT N$(X):NEXT
230 GOSUB 260
240 PRINT "THAT'S ALL"
250 END
260 VTAB 23:PRINT "PRESS RETURN TO CONTINUE";:
    GET R$:HOME:RETURN
270 DATA ONE,TWO,THREE,FOUR,FIVE,SIX,SEVEN,
    EIGHT,NINE,THN
```

变量简表

1. 变量名不应含保留字。变量名第一个字符应为字母,后边可跟237个字母或数字。

2. 变量名只有前两个字符在区别不同变量时有效。

3. Applesoft使用的变量有三种类型:

   a. 实型变量取值范围:从−9.99999999E+37到+9.99999999E+37。

   b. 整型变量,从−32767到+32767的整数,整型变量用标示符%表示。

   c. 字符串变量,含有255个英文字母和数字,用标示符$表示。

4. 可从键盘或从程序中向变量赋值。

INPUT 1. 此命令在屏幕上显示一个问号(?),让你

输入一个值，存入规定的变量。

2. 在一个 INPUT 语句中可以向多个变量赋值，多个变量之间隔以逗号。

INPUT 提示 1. 提示是在要求输入数据之前显示的信息。

2. 它告诉你应该输入哪种类型的数据。

3. INPUT 语句中的提示要用双引号括起来，用分号结束。

GET 1. GET 是从键盘接受一个字符，输入的这个字符屏幕不显示。

2. 回答完 GET 语句后不用按回车键。

3. 回答 GET 语句时，一定要赋值给字符串变量，以后需要时再转换。

4. GET 没有提示。若需提示信息时，可使用 PRINT 语句。

LET 1. LET 是一赋值语句，用来给定量赋值。

2. LET 把等号右边的变量或表达式赋值给等号左边的变量。

3. 程序中的每一个变量均可用 LET 语句多次赋值。

4. 只要程序中左边有一变量，后跟一个等号，即可使用 LET 语句。LET 一词可省略不用。

READ&DATA 1. DATA 建立一张数据表，供 READ 使用。

2. DATA 语句可以位于程序中的任何位置，可以累积。

3. DATA 语句中各数据项之间用逗号隔开。

4. READ 语句用来把 DATA 中的数据赋值给程序中的变量。

5. 每一个 READ 变量依次取数据表中的数据。

6. READ 每读取数据表中的一个数据，DATA 指针指向数据表的下一个数据。

7. RESTORE 语句将指针返回数据表的第一个数据，令 READ 重新读取 DATA 数据。

8. READ 超出 DATA 数据项时，出现 OUT OF DATA 出错信息。

变量类型的转换

1. INT(X)用来把实型数转换为整型数，去掉一个数的小数点部分。

2. LET 可用来把整型变量转换为实型变量。

3. VAL(X$)将字符串转换为实型或整型值。字符串 X$ 的第一个字符应为数字，否则字符串将转换为零。

4. STR$(X)或 SRT$(X%)把实型或整型变量转为字符串。

字符串处理 1. LEFT$(X$,n)取字符串 X$ 中最左边的 n 个字符。

2. RIGHT$(X$,n)取字符串 X$ 中最右边的 n 个字符。

3. MID$(X$,n)取字符串 X$ 中从第 n 个到最后结尾的所有字符。

4. MID$(X$,n,m)在字符串 X$ 中从第 n 个字符向右取 m 个字符。

5. LEN(X$)确定字符串 X$ 中的字符数。

6. 两个或多个字符串之间可以用加号(+)连接起来合并成一个字符串。

数组 1. DIM 命令可以把变量组成一个表，称为数组。

2. 若忘记确定数组的大小，Applesoft 自动设定最大下标为 10。

3. 下标用来标识数组中的每一个元素，确定其位置。下标用园括号括起来。园括号中的下标可以是数，数值变量或数表达式。

冯　静　编译自《NIBBLE》1989.9 期

公　告

原《苹果园》杂志聘任的杂志征订代理人，随杂志更名自然转为《中华学习机》杂志的第一批征订代理人，现重新公布如下：

姓名	地址	邮政编码
马开云	天津碱厂机械车间(转)	300450
吴安栋	济南林祥南街 166 号	250001
王玉主	无锡市蠡园水秀饭店	214075
景成循	上海市普陀区曹杨五村 436 号	200062
倪建华	上海市南汇县新场乡中心校	201314
杜永平	四川省万县市师专服务部	634000
杜华光	大庆市林源炼油厂中学	163813
俞　鹏	南通市如东县教育局	226400
孟时成	浙江省绍兴教师进修学校	312000

《中华学习机》杂志社第二批征订理人如下：

姓　名	地址	邮政编码
何海群	长沙市公安专科学校	410006
许文平	西安市交通大学 8817 号	710049
陈兆前	浙江大学 58 信箱(10 舍—213)	
倪秀峰	上海市东汉阳路 246 号	200080

本刊希望有更多的征订代理人

面粉生产指标微机实时监测系统

济南北山面粉厂　于范

粮食行业中面粉加工是其主要的工业生产环节，而面粉生产有着若干个经济技术指标，这些指标互相制约，偏废任何一个指标都是不经济的。几十年来，在整个面粉生产过程中，信息反馈迟缓，全凭操作工人的眼观手感等感官来操作，随着科学技术的发展，迫切需要现代化的监测控制系统来指导现场生产，以尽快找到最佳经济点。

SMWJ——A 面粉生产指标微机实时监测系统就是一个操作指导控制系统。它的主要任务是采集现场信息(面粉、麸皮袋数和电度表转数等)，然后进行加工处理，并将结果信息及时显示和打印出来，使操作人员可以随时了解生产的瞬间变化趋势，提高操作的准确性，较快地找到最佳经济点，提经济效益，因此被称之为操作工人的“眼睛”。

一、概述

在 TP801—B 单板机的基础上，对输入输出通道和系统软件重新设计，组成了适合面粉加工的工业控制机。单板机的三总线通过三态缓冲器与外部三总线相连。模拟输入量通过隔离放大进入外部三总线。开关量、输入/输出量均采用光电隔离，抑制干扰，保证安全。数据传送采用程序查询和中断相结合的方法，软件按模块化结构设计，在管理程序调度下运行。

二、硬件结构(见图一)。

该系统由主机、打印机、控制电路板、显示屏、电源等部分组成。

1. 主机：选用 TP801— B。Z80—— CPU，EPROM2716(10K)，RAM6116(6K)，Z80—PIO—片，Z80—CTC—片，时钟频率为 2MHZ。

存储器地址分配如下：

地址	用途
0000H～07FFH	监控程序。
O800H——0FFFH	应用程序。
1000H——17FFH	应用程序。
1800H——1FFFH	控打程序。
2000H——27FFH	应用程序工作区。
2800H——2FFFH	
3000H——27FFH	字库打印格式。

3800H——3FFF(2KSRAM6116)监控程序工作区(占96字节)。

2. 键盘与显示：

(1)键盘：在线使用的专用键盘，十三个命令键，一键多用。它决定七种生产方式，四种打印格式，调整年、月、日、时、分、秒，键的状况通过 74LS244 缓冲器进入主机，占用两个口地址，主机查询键的状态，以判断需要执行什么操作。见图 2 键盘原理图。

(2)显示：把需要显示的生产指标数值由 74LS273 输出，经过驱动器加到七段 LED 数码管上，用 16 个 LED 数码管分四组，一次显示四种生产指标及其相对应的数值。而又将 16 项生产指标分为四组，以轮流分时显示的方式，在五分钟内显示两次，全暗一分钟，以区别刷新后五分钟内的新的数值。每送出一组指标时，显示器上以相应的汉字指标名称和单位显示出来，生产指标数值的显示用每组四个共四组的 LED 数码管，设置在显示器的下方，占用 16 个口地址，汉字显示设置在显示器的中间部位，占用一个口地址，用数据字确定是哪一组的汉字，确定之后，便接通指标显示灯，则亮，相对应的生产指标的汉字就通过灯光投射显示在显示器上，一目了然。

时钟显示由六个 LED 数码管组成，设置在显示器的右上角，运行前可以校对年、月、日，运行过程中校对和显示时，分、秒。

3. 采样：其原理框图见图 3

通过传感器以五分钟为一个周期，把现场面粉打包机的面粉袋数和麸皮打包机的麸皮包数及有关的电度表的转数等信号转变为开关量的电信号采集来，经过光电耦合器隔离、整形，由 PIO 进入计算机。计算机的 PIO 的 A 口置为输入方式，为了满足打印要求，B 口置为位控方式。每五分钟刷新一次。

五台打包机的信号线接到 PIO 的 P A_0 一 P A_4 上，两台麸皮打包机的信号线接到 P A_5 、P A_7 上，五只电度表的信号线接在 PIO 的 P B_0 一 P B_4 上。七种生产方式是由键的状态和软件控制来完成的。如生产特二粉、次粉，则按下键 K_0 、K_5 、K_7 ，且按下相应的班次(如甲班)键 K_5'，则甲班在生产上述二种面粉。其他生产方式和打印方式等按下相应的键即可。本设备灵活、简便。

4. 打印：

打印机选用 TP801P—24 点阵式微型打印机，每行印 24 个 5×7 点阵的字符。可以打印字符、图案和少

量汉字。这种打印机具有"节拍式限时中断"方式，在这种方式下打印回程时输出一个"RETURN"信号。该机不带 CPU，由主机 CPU 执行控打程序，控制打印机的工作，因此在打印机工作时，主机 CPU 不能执行计时采样和运算程序等，为此把"RETURN"信号接到 PIO 的 PB 上，且把 B 口置为位控方式，这样打印机在打印完成一点行返回时，请求中断，执行中断服务程序，完成采样、计算和计时等项工作，使系统保证正常工作。

三、软件部分

（一）组成：整个系统由 3700 多条命令组成。其主要程序为：

1. 键盘整询程序；
2. 采样程序；
3. 计算主程序；
4. 显示主程序；
5. 打印程序；
6. 自诊断程序；
7. 计算打印程序；
8. 采样处理程序；
9. 打印中断服务程序；
10. 厘米时产量程序；
11. 时钟程序；

还有 28 个子程序，两个字模表，自定义 26 个汉字，如：小时，面粉、产量、机、袋、本、特、标、次、麸皮、比、电、耗、吨、厘、米、磨、白、度、上、班、包。

整个软件功能强、灵活，使硬件显著节省，从而使该系统成本降低。

（二）主要程序框图：

文中绘出主程序流程图，(图 4)计算程序流程图(图 5)，其他流程图略。

（三）该系统功能

1. 采集开关量：本系统为十二个(可由需要而定)。
2. 采集一个模拟量。
3. 确定生产方式：七种。

打包机号 / 生产品种 / 生产方式	1#	2#	3#	4#	5#
1.500	特	特	特	特	特
2.050	标	标	标	标	标
3.005	次	次	次	次	次
4.221	特	特	标	标	次
5.140	特	标	标	标	标
6.302	特	特	特	次	次
7.041	标	标	标	标	次

注：　特——特一粉或特二粉

标——标准粉

次——次粉(低于标准粉的面粉)。只要按下相应的键即可达到所要求的生产方式。

4. 显示：输出显示十六项生产指标。

动态指标十二项			静态指标四项
特粉产量(袋/时)	特粉麸比%	电耗 KWh/h	面粉总产量袋/时
标粉产量袋/h	标粉麸比%	吨电耗 KWh/T	麸皮产量 袋/时
次粉产量 袋/h	次粉麸比%	磨机电耗 KWh/h	总粉麸比%
麸皮产量 袋/h	白度%	厘米时产 kg/cmh	小时电耗 KWh/h

显示时，其顺序自上而下分四组(即四行)轮流显示，每组(即每行)显示时间为 30 秒。

动态指标反映生产趋势，刷新周期为 5 分钟；静态指标显示上一小时的实际累加值，刷新周期为一小时，为均衡生产提供数据。

5. 实时时种：在校正时，时钟显示年、月、日，或时、分、秒；在正常进行时，显示时、分、秒。

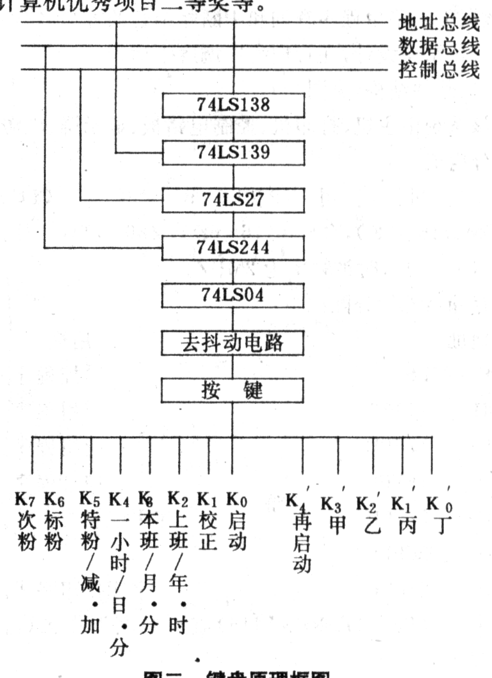
6. 打印：方式为四种。每种 14 项生产指标。可以打印：本班、上班、一小时、八小时的生产指标。前三种为随机打印，后一种为自动打印。

7. 人机联系：

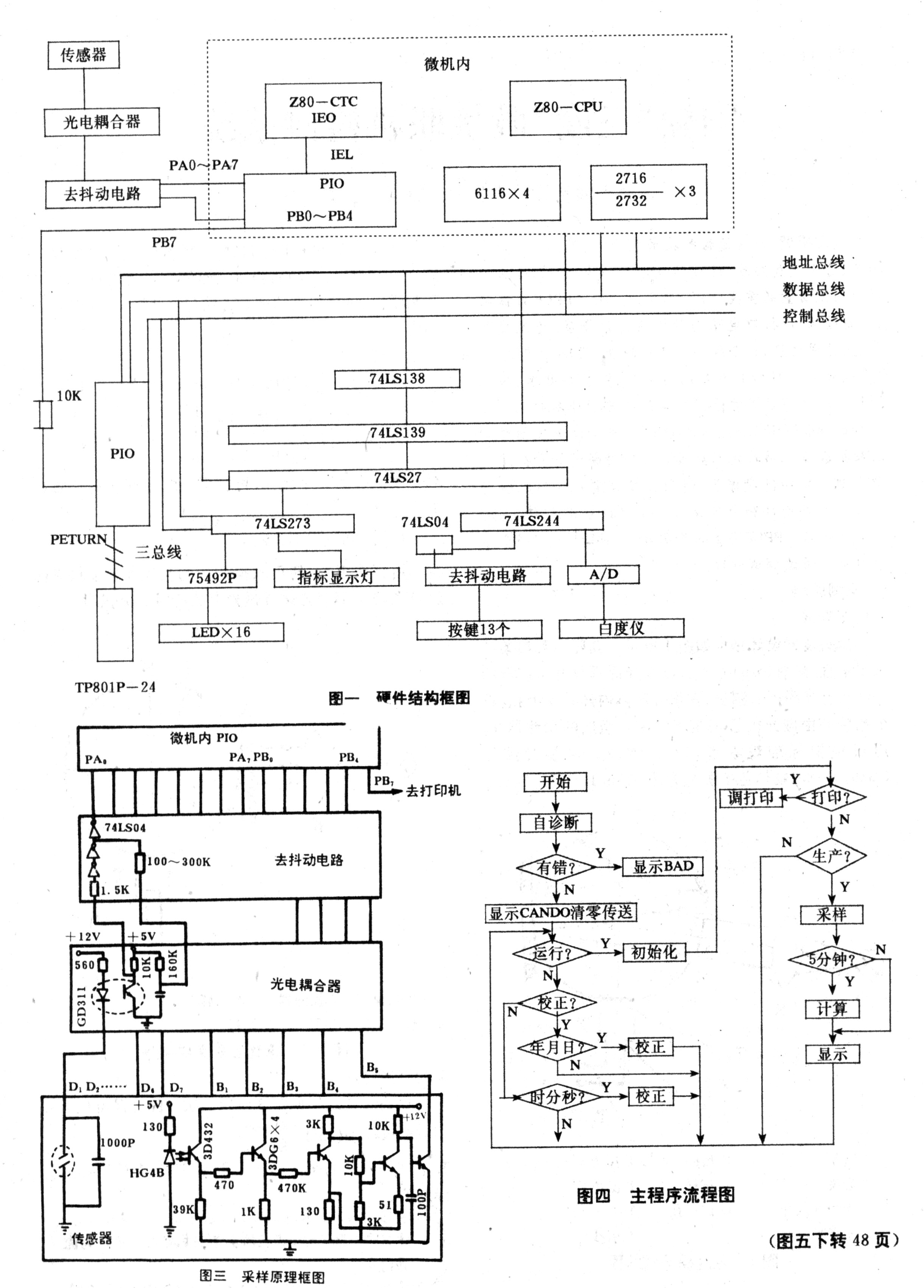
设有一个专用控制键盘，用于完成时间校正，确定生产班次、选择生产方式和打印方式等。

四、经济效益：

该装置具有功能较强、操作方便、性能价格比高、结构紧凑、体积小等优点。自 1987 年以来相继在多个厂家推广应用，均获得较好的经济效益和社会效益。曾先后获得中华人民共和国商业部科技进步四等奖和省计算机优秀项目二等奖等。



图二　键盘原理框图



图一 硬件结构框图

图三 采样原理框图

图四 主程序流程图

(图五下转 48 页)

陶瓷“三度”微机联机检测系统*

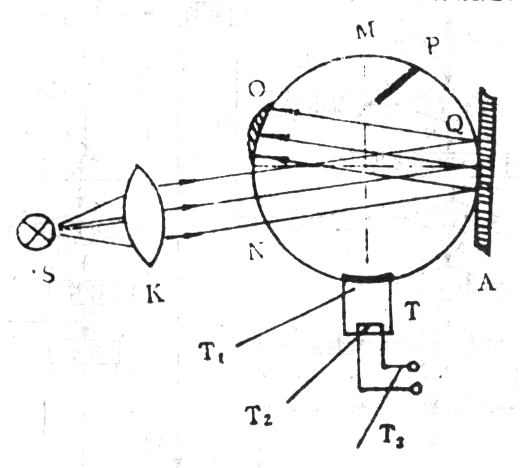
江西省陶瓷质量监督检验站　　吴　坚

内容提要：在陶瓷检测技术领域中，陶瓷产品的光泽度、透光度和白度（统称为陶瓷“三度”，下同）是三个评价陶瓷质量的重要指标。本文介绍在 APPLE II 微机上，通过 A/D 接口电路与外设陶瓷光泽度、透光度和色度（适用白度）三个探头装置的联机，实现对陶瓷“三度”试样的分时检测数据的自动采样、数据处理、报表打印和数据存储等方法。本系统测定方法参照采用了现行的国标方法，并结合数理统计理论给出较为科学的表报数据。同时还研究设计了一种新型的积分球式透光度探头和色度探头（适用白度测定），使其比现行采用的透光度计和白度仪有较高的精确度和科学性。本系统已在 APPLE II plus 型微机和 XMF——I 型中华学习机上通过调试运行。

一、检测原理

1. 色度测试

探头装置设计结构如图 1 所示。其照明观察条件采用垂直/漫射（o/d）方式。其中，光泽吸收并可进行消光泽和有光泽下的测量；在积分球体内开有 4 个孔，每个孔装有由硅光敏管和相应滤色片组成的探测器 T，用于模拟标准观察者 X_1、X_2、Y、Z。该装置满足 GB3979—83（物体色的测量方法）第 5.1.4 条规定。



S 光源　　P 挡板　　T 探测器
K 透镜　　A 试样　　T_1 滤色片
N 光线导入孔　　Q 试样孔　　T_2 P—N 区
O 光泽吸收井　　T_3 引线

图 1　色度探头结构图

通过在系统中加入滤色片，模拟一定标准光源下的 $\overline{X}(\lambda)$、$\overline{Y}(\lambda)$、$\overline{Z}(\lambda)$ 而获得试样色反应函数的 X、Y、Z 三刺激值。试样白度的测试方法与之相同，但描述方法不同。

2. 透光度测试

探头装置设计结构如图 2 所示。试样的透射率为：

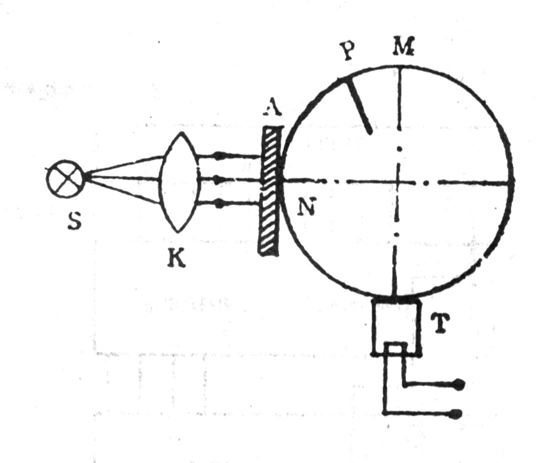
$$\tau=\frac{It}{Io}=\frac{Kl-K'}{Ko-K'}$$

$$\log(It-Io)=-\varepsilon d+b$$

对于某试样的 n 个坯体，可测得一组厚度 d 不同的数据。

$$(\frac{It}{Io})_1, dl, (\frac{It}{Io})_2, d2, \cdots\cdots, (\frac{It}{Io})_n, dn$$

用直线拟合，便可求得相应的消光系数 ε 和表面反射系数 b，而获得试样透光度的相对百分强度。



S 光源　　T 探测器　　A 试样
K 透镜　　N 入射孔　　R 挡板

图 2　透光度探头结构图

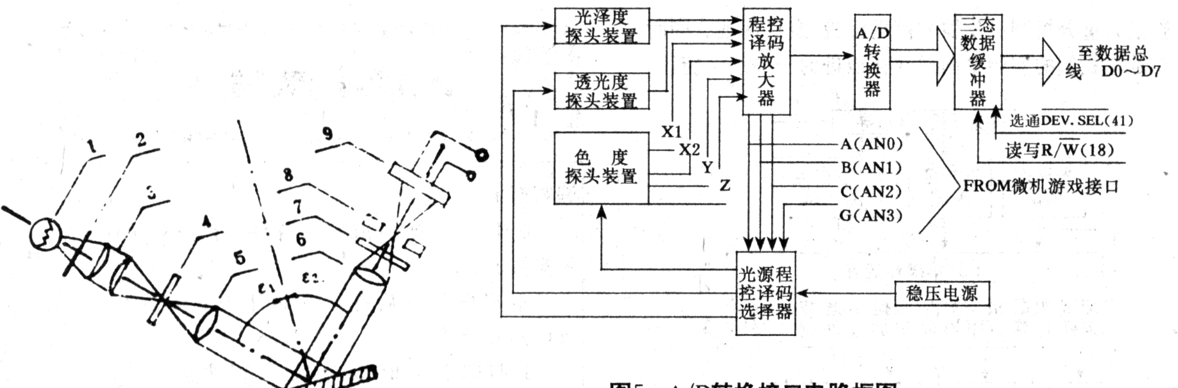
3. 光泽度测度

探头装置结果如图 3 所示。试样的光泽度值为：

$$Gs(45°)x=Gs(45°)o\frac{Kx-K2}{Kl-K2}$$

其中试样测试的平面最大反射率，表示为该试样的光泽度值。

* 此文在中国 Apple 微机协会第三届年会上被评为优秀论文



1、光源　2、升色温滤色片　3、聚光透镜组
4、光源光栏　5、入射透镜　6、接收透镜
7、视见函数修正滤色片　8、接收光栏
9、光接收器　$\varepsilon_1=\varepsilon_2=45°$

图 3　镜向光泽度探头结构图

二、系统硬件配置

系统硬件配置如图 4 所示。稳压电源采用抗扰动的无级平滑稳压电路。其设计参数为：

电源电压：220V±10%，50/60Hz

输出电压：6V，9V

最大输出电流：5A

电压调整率：0.005%

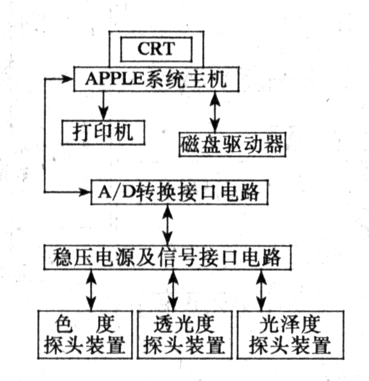


图 4　系统组成框图

电流调整率：0.1%

工作温度：0～60℃

信号驱动电路是陶瓷“三度”探头装置的选通切换继电器电路，主机通过游戏 I/O 接口的 AN0/AN3 开关信号线，发出控制选通信号，由外设槽 A/D 转换接口卡上的译码器选通输出，使对应的继电器吸合/断开，接通/关闭相应的探头光源。A/D 转换电路如图 5 所示。

图5　A/D转换接口电路框图

三、系统软件设计

陶瓷“三度”微机检测系统的软件，分外部介面和内介面两个方面进行分析设计。外部介面的设计以系统面向用户的实用性及其改善 CRT 屏幕视读操作为主，内部介面的设计则以程序积木结构的灵活性和提高系统采样速率为主。

1、外部介面设计

包括：①分别采用闪动光标、条形光标和流动光标以便对各阶段当前的操作状态作区别和醒目提示。例如在试样测试时，在数据表域中的数据栏位上设定一个相应的(可移动)条形闪动光标指针，指示当前采样窗口的测试数据将待存放的位置。②采用主、次菜单的层次和嵌套结构形式，且用户只需按动菜单提示键，系统即可进入相应的操作、处理状态或返回。同时对其它的误键动作，系统均与屏蔽。③将屏幕分成若干个显示窗口：标题部、数据表域、菜单反显窗口以及操作提示和动态采样窗口，以便易于用户的视读操作。④参照采用现行使用的陶瓷“三度”国标测定方法，并同时计算出各检测数据的数理统计值，以便进行检测对比和综合评价。

2、内部介面的设计

包括：①程序设计采用积木结构模块的形式，这对于模块的设计装配和调试，主程序中关于标准参数的修正，都较方便。②程序采用 BASIC 语言和 6502 机器语言编写。机器语言程序总占用系统的 $9000～$95FF RAM 内存空间，其中包括了测试采样、帧画面图形、主菜单选择指针、键盘搜索等子模块，可以提高采样速率和画面切换效果。

主程序设计框图如图 6 所示。图 7 所示是一例光泽度测试主菜单屏幕的帧显示画面。

四、小结

将微机(尤其如 APPLE II 和中华学习机此类价廉的低档机)应用于检测技术中，实现多种(或多台)分析测试仪器联机检测的一机多能化，对改造我国目前实

验室大量分析测试仪器的落后面貌，意义深远。同时，亦是使我们的实验室向现代化迈进的一条经济捷径。

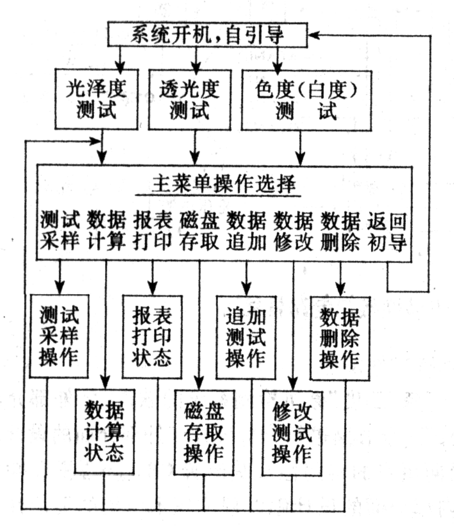


图6：程序设计框图

PLOSSINESS TEST

	G1(X)	G2(X)	G3(X)	G4(X)	G5(X)	Gm(X)
(1)	72.3	72.1	69.5	74.0		74.0
(2)	68.4	65.5		66.6	66.5	68.4
(3)	72.7		71.6	71.3		72.7
(4)	69.8	67.7	66.3	68.8	69.5	69.8
(5)		77.5	77.2	77.6		77.6
(6)	75.6	75.4		75.8		75.8

1⇒TEST 2⇒COUNT 3⇒LPRINT 4⇒DOSR/W 5⇒APPEND 6⇒EDIT 7⇒DELETE 8⇒RSTART

〔*〕:press〔RETURN〕to show accept this
press〔SPBAR〕to show choose next

图7： 光泽度测试主菜单帧屏幕

敬告读者：

凡欲参加中国中华学习机普及协会者，请直接与协会办公室联系。

协会地址：北京市西三环北路甲45号

邮政编码：100081

·读者来信·

编辑部负责同志：

《苹果园》杂志办的很有特色，有些文章有很大的实用价值。

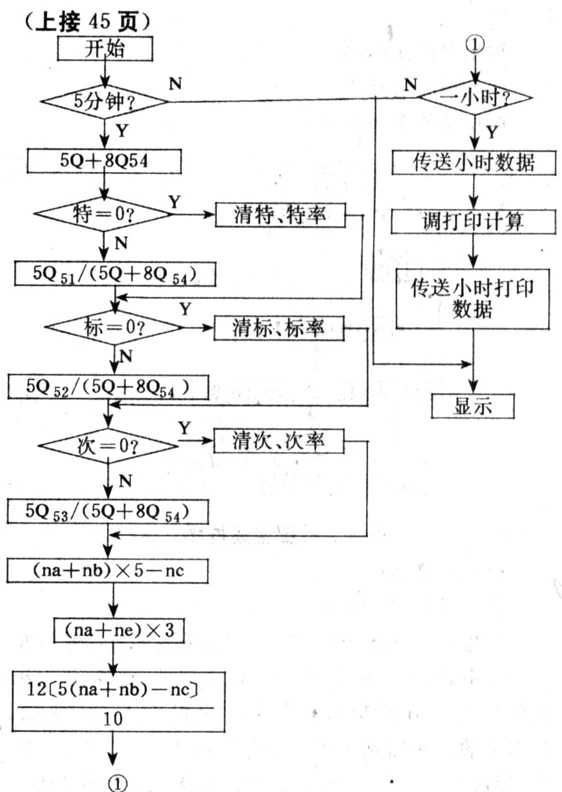
希望贵刊今后能登载以下有关中华学习机CEC—I的文章：

1. 打印卡、EPROM写入卡、存贮器扩充卡、Z80卡、80列卡、可编程音乐合成器卡、A/D卡、RS——232卡等硬、软件。
2. 怎样进行动画设计，如何编写游戏程序。
3. APPLE II与中华学习机CEC——I在硬、软件上，有何异同。
4. 中华学习机的维修、改进、使用技巧、经验交流等。
5. 刊登高质量的游戏、教学、管理等程序。

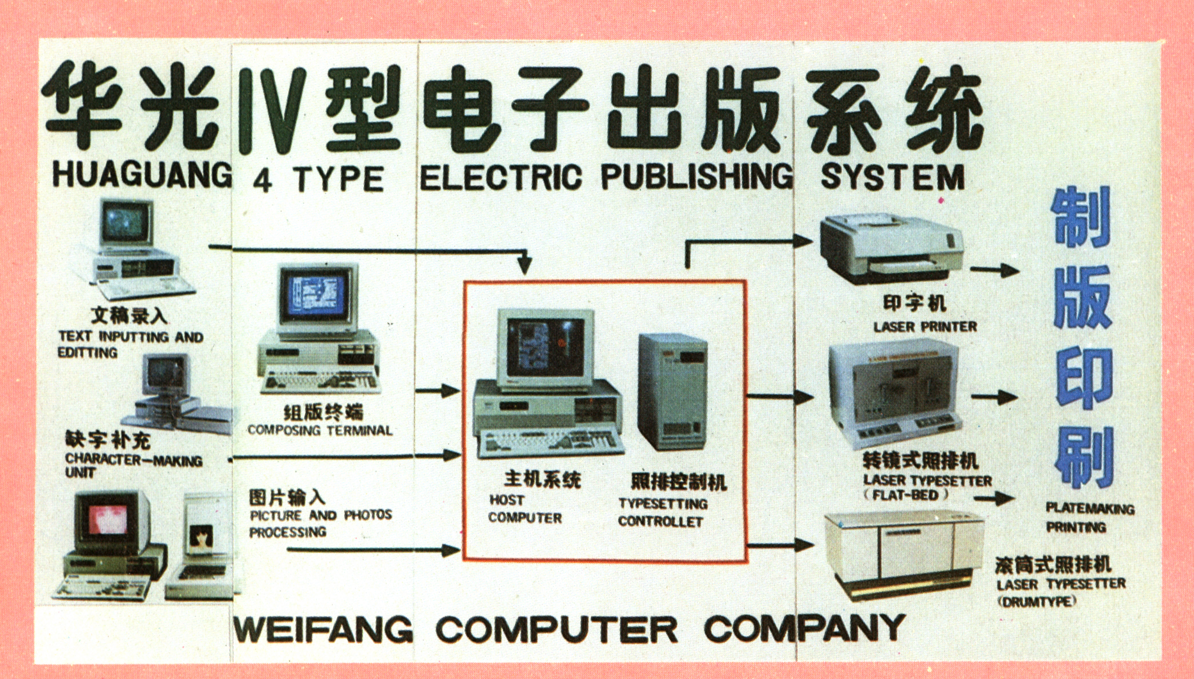
读者：陈永强

1990.元.8

(上接45页)



图五 计算程序流程图



华光®

华光IV型电子出版系统是潍坊计算机公司奋战十几年的高技术成果和中华民族文化的结晶。

华光，是潍坊计算机公司的产品注册商标。

国家华光电子出版系统的生产基地在潍坊。

华光电子出版系统自1985年通过国家级鉴定以来，荣获多项国内外大奖和多次在世界银行招标中标，1989年又独家获得国家质量银奖。

华光电子出版系统最新成果有： ·高集成度、高可靠性、高功能照排控制机TC——89。 ·国内最新型全版面显示，多功能大屏幕报纸、广告处理系统 ·最新报版、书版、杂志版合用编辑系统 ·高清晰度、多功能图片照片处理系统 ·朝文、维文、蒙文、傣文、藏文、阿拉伯文等多种文字处理系统

华光用户已遍及全国(含台湾)及东南亚地区。华光电子出版系统是印刷出版界朋友的忠诚伙伴。

